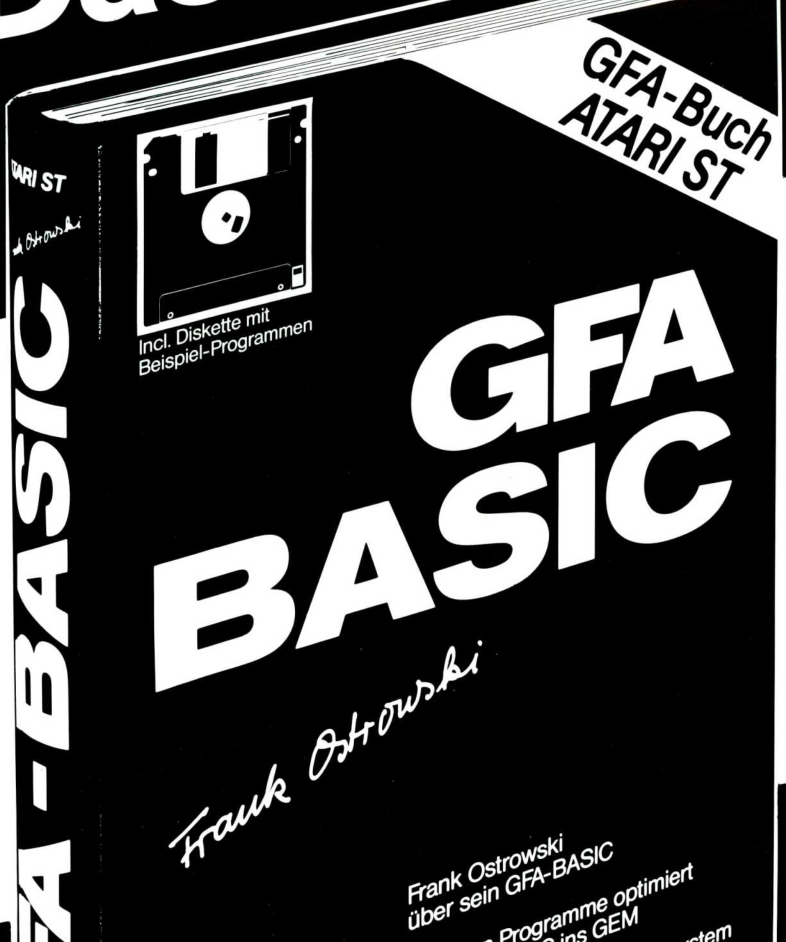


ATARI
ST COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
Februar '87
ÖS 53,—/Sfr. 6,—
DM 6,—
2
Der ST als Macintosh
Zwei Emulatoren im Vergleich
GFA-Basic-Compiler
Wie schnell ist er wirklich?
STAD
ST Aided Design
HITS von ATARI
* MS-DOS-Rechner
* Die neuen Modelle und Preise

Das Original



GFA-Buch DM 79,–

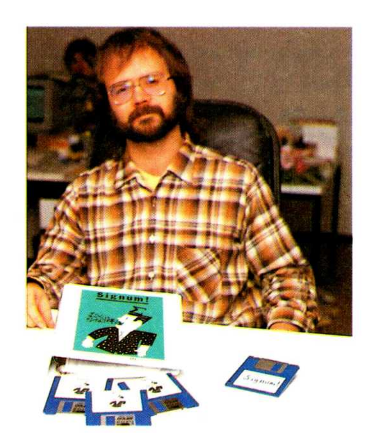
...Anruf genügt: 02 11-58 80 11

Lieferbar
voraussichtlich
Ende Februar 87

GFA Systemtechnik GmbH

Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 02 11/58 80 11

GFA SYSTEMTECHNIK



Aktualität um jeden Preis?

Liebe Leserin, lieber Leser,

Original, Vorabversion oder gar Raubkopie? Diese Frage stellt sich in unserer Redaktion bei jeder Ausgabe dieser Zeitschrift neu.

Das Problem mit den Raubkopien ist bekannt: Aus zweifelhaften Quellen wandern viele Produkte von Hand zu Hand, noch bevor sie überhaupt auf dem deutschen Markt ausgeliefert wurden. Meist sind es Programme, die zwar bereits in den USA und in Großbritannien erhältlich sind, aber entweder noch eingedeutscht oder den deutschen Zoll noch nicht passiert haben.

Ähnlich problematisch sind die Vorabversionen. Sollte man ein Programm schon besprechen und kritisieren, loben oder gar „verreißen", ohne die endgültige Version in den Händen gehabt zu haben?

Aktualität vor Qualität? Schnell mal in die Anleitung geschaut und einen Testbericht geschrieben – oder das Programm gründlich ausprobieren und dann ausführlich berichten?

Wir meinen, der Qualität sollte in jedem Fall der Vorrang eingeräumt werden – sogar dann, wenn die Aktualität darunter leiden sollte. Was nützt der schnelle Testbericht, wenn in der Eile nicht genügend Zeit zum ausgiebigen Testen war? Deshalb müssen wir leider manchmal den einen oder anderen Bericht auf die nächste Ausgabe verschieben. Diesmal ist es der Art- und Filmdirector, der uns erst kurz vor Redaktionsschluß erreichte.

Zudem sind wir entschlossen, keine einzige Raubkopie zu testen. Schon deswegen sind wir darauf angewiesen, erst einmal die Originale auf dem Tisch zu haben. Es ist ja auch schon vorgekommen, daß Original und Raubkopie unterschiedlich waren.

Ebenso wichtig wie Aktualität sind für uns Qualität der Berichterstattung und die Fairneß gegenüber den Programmautoren. Für unsere Haltung in dieser Frage hoffen wir auf Ihr Verständnis.

Harald Egel

STAD – mehr als nur ein Zeichenprogramm!

Mit STAD (ST Aided Design) zeigt sich einmal mehr, daß jetzt ca. 1 1/2 Jahre nach Markteinführung des ATARI ST in Deutschland immer mehr professionelle und preiswerte Software auf den Markt kommt. STAD beinhaltet einen 2D-Zeichenteil mit eigenem Zeichensatzeditor und einen 3D-Konstruktionsteil, mit dem sich Rotationskörper und Drahtgitter spielend erstellen lassen. Mit seinen vielfältigen Möglichkeiten ist STAD in einem Testbericht nicht annähernd so vorzustellen, wie es eigentlich seinen Leistungen gerecht werden sollte, doch lesen Sie selbst.

Macintosh Emulatoren im Vergleich

Wurde bisher nur von einem Gerät gesprochen, daß ermöglichen soll, Programme des APPLE Macintosh auf dem ATARI ST zu emulieren, so bieten wir Ihnen einen Vergleichstest zwischen zwei solcher Geräte. Das Erste nennt sich MAGIC SAC, stammt aus amerikanischer Produktion und ist schon überall in der Werbung der Soft- und Hardwarefirmen vertreten. Das zweite Gerät kommt jetzt neu auf den Markt, trägt den Namen ALADIN und wurde im Karlsruher Raum entwickelt. Was beide können und ob sie wirklich so kompatibel zum Macintosh sind, erfahren Sie in unserem Testbericht.

Der Compiler für das GFA-BASIC

Der GFA-BASIC-Interpreter ist nicht nur wegen seiner Struktur eine BASIC-Revolution (keine Zeilennummern, Programmstruktur, etc.), sondern auch wegen der für einen Interpreter undenkbar schnellen Ausführung. Wie kann man eine solche Sprache noch erweitern? Dafür gibt es jetzt einen Compiler. Wir testen für Sie den brandneuen Compiler für den GFA-BASIC-Interpreter.

Allgemeines

Software

Hardware

Grundlagen

Kurse

Aktuelles

HITS '87

Neue ST-Modelle mit 1, 2 und 4 MByte Speicherkapazität

Auf der CeBit in Hannover werden von ATARI drei neue Versionen des ST vorgestellt. Es handelt sich dabei um Rechner, die mit 1, 2 oder 4 MByte Speicherkapazität ausgerüstet sind und in einem PC-ähnlichen Gehäuse untergebracht sind, das in seiner Grund fläche kaum größer ist als der monochrome Monitor. Das Gehäuse besteht aus zwei „Kästen" mit den Maßen von jeweils ca. 34 x 34 x 5 cm (BxLxH), die aufeinander gestellt werden. Im unteren Gehäuse ist der Rechner mit Netzteil untergebracht und im Oberen die Floppy Laufwerke bzw. eine Festplatte. Die Tastatur ist in einem separatem kleinen Gehäuse integriert, das über ein Spiralkabel mit dem Rechner verbunden ist. Sie sind voll kompatibel zu den derzeitigen ST-Modellen.

Ein Laserdrucker für die ST-Modelle

Ebenfalls wird auf der CeBit ein preiswerter **ATARI Laserdrucker** für die ST-Modelle zu sehen sein. Der Drucker besteht größtenteils nur aus der Mechanik, die Steuerelektronik übernimmt der Rechner. Dadurch läßt sich der günstige Preis von etwa DM 3000,– (ohne Gewähr) erreichen.
Selbstverständlich werden wir in den nächsten Heften über diese Sensationen noch ausführlich berichten.

Neue Preise bei ATARI!

Bei dem ST-Modell **ST/F 1040** wurde erneut der Preis gesenkt, so daß man ihn nun zum Preis von nur **DM 1998,–** mit monochromen Monitor erhalten kann. Ohne Monitor kostet er DM 1598,–.
Ebenfalls billiger geworden ist das doppelseitige Laufwerk **SF 314**, das jetzt nur noch **DM 598,–** kostet.
Weiterhin ist der Preis für den monochromen Monitor **SM 124** von DM 598,– auf **DM 498,–** gesenkt worden.

1st Word Plus

Das schon für Ende '86 angekündigte Textverarbeitungsprogramm 1st Word Plus ist wegen mehrerer kleinerer Fehler im Programm **gestoppt** worden.
Ob und wann es auf den Markt kommt bleibt abzuwarten, schade!

ATARI Corp. (Deutschland) GmbH
Frankfurter Straße 89–91 · 6096 Raunheim

News

Die flexible 2 -Gate-Array-Familie

Mit den 2 -HCNOS-Gate-Arrays der Serie HCA62A00 stellt Motorola eine neue Familie applikationsspezifischer ICs vor, die sich durch uneingeschränkte Flexibilität bei der Konfiguration der I/O-Struktur und bei der Anordnung der Stromversorgungs-Anschlüsse auszeichnet. Semicustom-Produkte werden damit einfacher in der Anwendung, weil jeder einzelne Anschluß als Ein- oder Ausgang, als bidirektionale Leitung oder als Versorgungsspannungs- oder Masseanschluß definiert werden kann. Die neue Produktfamilie wird in sieben Versionen mit 600 bis 8500 Gattern angeboten.

Das Leiterplattenlayout und die Umsetzung eines Schaltplans in einen Semicustom-Chip werden durch die große Flexibilität der Serie 62A00 bedeutend vereinfacht, weil jeder Anschluß so konfiguriert werden kann, daß sich der geringste Aufwand ergibt.

Motorola GmbH
Arabellastr. 17
Postfach 81 06 20
8000 München 81
Tel.: 0 89 / 92 72-0

HCA62A00 SERIES
CMOS MACROCELL ARRAYS

Universal I/O Buffers
Output Enable A
Output A
Input A
2X Output Drive A
Bonding Pads
Input B
B
PLCC
LDCC

PARALLELED BIDIRECTIONAL BUFFER WITH 2X OUTPUT DRIVE AND FULL PIN UTILIZATION

MOTOROLA ANNOUNCES THE INDUSTRY'S MOST FLEXIBLE FAMILY OF 2-MICRON GATE ARRAYS

BITZEPS

Das Softwarehaus RÜGER & GODEHARDT bietet unter dem Namen **Bitzeps** ein Softwarepaket an, das die komplette Verwaltung eines Trainings-Studios ermöglicht.

Neben einer ausgezeichneten Mitgliederverwaltung erledigt dieses Programm Ihren gesamten Schriftverkehr wie etwa Lastschriften, Mahnungen, Geschäftskorrespondenz oder Vertragsverlängerungen. Zu dem Programmumfang gehört auch eine Vielzahl bewährter Trainingssysteme, die dem Anwender auf Knopfdruck zur Verfügung stehen.

Die graphische Bedieneroberfläche ist angenehm für Anwender ohne Computererfahrung.

Rüger & Godehardt
Adlerstr. 41
4600 Dortmund 1
Tel.: 02 31 / 14 75 31

ECB und VME Bus-Interface für ATARI ST Rechner

Zwei verschiedene Interfaces werden von der **GTI – Gesellschaft für technische Informatik** in Berlin für den ATARI ST angeboten. Eine ECB-Bus-kompatible Interface-Karte ermöglicht die Anwendung dieses Rechners im professionellen Bereich der Meß- und Regelungstechnik. Der Anschluß an den ATARI erfolgt über den DMA-Port, ohne daß der Rechner geöffnet oder in irgendeiner Weise modifiziert erden muß. Eine Parallelbuchse erlaubt den gleichzeitigen, unbeeinflußten Betrieb der ATARI-Festplatte am DMA-Port. Auf dem ECB-Bus stehen ein Adreßraum von 8 MByte oder sieben Bänke zu je 64 kByte zur Verfügung. Alle Signale der Interface-Karte folgen dem ECB-Standard, so daß auch ein Einsatz in Multi-Master-Systemen möglich ist. Zum Lieferumfang gehören neben der Interface-Karte ein ausführliches deutsches Handbuch sowie eine Diskette mit Treiber- und Beispielprogrammen. Der Verkaufspreis beträgt DM 498,–.

Für Anwender mit höchsten Ansprüchen an einen Sytembus wird eine VME-Bus-kompatible Interface-Karte angeboten. Der Anschluß an den ATARI erfolgt ebenfalls über den DMA-Port. Auch hier erlaubt eine Pa-

Sonderheft

für alle ST-User



Das ST-Sonderheft bietet unter anderem folgende Themen:

- **Compilertechnik leicht gemacht**
  Die komplette Anleitung zum Programmieren eines Pseudo Pascal Compilers mit Grundlagen und Technik
- **Einführung in Prolog**
  Alles über die Sprache der künstlichen Intelligenz Prolog. Eine Einführung für Einsteiger und Fortgeschrittene mit vielen Beispielen.
- **3 D Graphik**
  Ein Programm zur Erstellung von Rotationskörpern. Diese Körper kann man konstruieren, vergrößern, verkleinern und rotieren lassen.
- **Trickfilm**
  Ein Programm zur Erzeugung von kontinuierlichen Bewegungsabläufen.

- **Superschnelle Apfelmännchen**
  Erzeugen bis zur Ausschnittsvergrößerung ist kein Problem. **Der Computer als Psychologe:** Mit Eliza, dem Programm für künstliche Intelligenz. **Eine Super 68000 Assembler Bibliothek:** Routinen einfach aufrufen und effizient programmieren. **Ray Tracing Programm:** Mit diesem Strahl-Verfolgungsprogramm erzeugen Sie in normalen Bildern (decas) phantastische Graphiken und Spiegelungen (s. Titelbild). Sowie Bericht über **Monitore und Drucker** für den Atari und über den Stargast **Commodore Amiga.**
- **Fensterverwaltung**
  Ein Superprogramm in **GFA-BASIC.**

Bestell-Abschnitt

Einsenden an: Heim-Verlag · Heidelberger Landstr. 194 · 6100 Darmstadt-Eberstadt

Senden Sie mir bitte ______ Stück des ST-Sonderheftes

Ich zahle keine Versandkosten. Den Betrag für das Sonderheft begleiche ich durch

☐ beigefügten Verrechnungsscheck
☐ beigelegte Briefmarken

Name

Straße

rallelbuchse den gleichzeitigen, unbeeinflußten Betrieb der ATARI-Festplatte. Auf dem VME-Bus steht der volle 16-MByte-Adreßraum zur Verfügung. Der Einsatz in Multi-Master-Systemen wird durch eine integrierte Bus-Arbitrations-Logik ermöglicht. Zum Lieferumfang gehören neben der Interface-Karte ein ausführliches deutsches Handbuch sowie eine Diskette mit Treiber.

Ferner bietet die Firma **GTI** unter dem Namen IsGemDa ein Datenbanksystem für den ATARI ST an. IsGemDa ist ein Dateiverwaltungs- und Datenbanksystem, das sich durch eine optimale Bedienerführung auf der Grundlage der graphischen Benutzeroberfläche GEM auszeichnet. Das Programmpaket gliedert sich in zwei Teilprogramme, die jeweils selbständig ablauffähig sind.

Mit dem DESIGN-Programm kann sich jeder Anwender seine persönliche Datenbank gestalten. Die Möglichkeiten reichen vom Entwurf der Datenstruktur über freie Bildschirmmasken mit graphischen Objekten, Listen und Druckformularen bis hin zur Auswahl anwendungsspezifischer Menüs und Pictogramme (Icons).

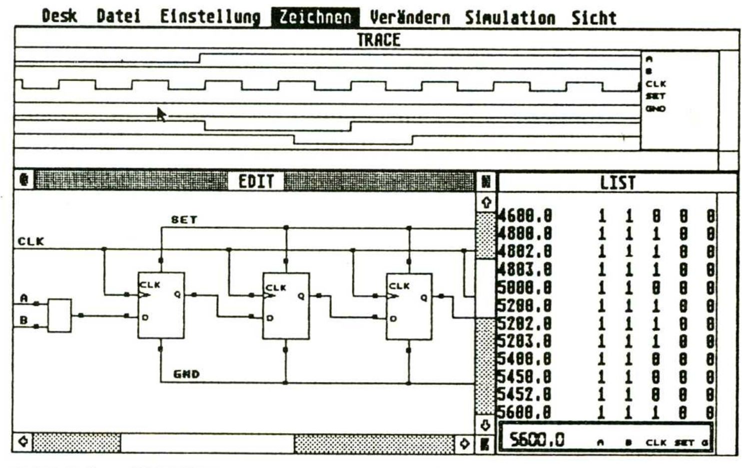
Mit der so gestalteten Datenbank kann dann mit Hilfe des IsGemDa-Programmes gearbeitet werden. Hier werden Daten erfaßt, geändert, gelöscht, gesucht und ausgegeben. Selbstverständlich kann IsGemDa mit den verschiedensten Datenbeständen arbeiten, die mit dem DESIGN-Programm entworfen wurden (bis zu acht gleichzeitig).

Beide Programme werden sehr komfortabel mit der Maus bedient und enthalten umfangreiche Hilfstexte, die für drei Erfahrungsstufen eingerichtet sind (Anfänger, Normal, Experte). Die Anwendungsmöglichkeiten von IsGemDa erstrecken sich von jeder Form der Datenverwaltung, sei es nun eine einfache Kunden- oder Lagerkartei, bis hin zum komplexeren Informationssytem.

Ein komfortables deutsches Handbuch gehört zu dem Lieferumfang.

GTI – Gesellschaft für technische Informatik mbH
Unter den Eichen 108 a
1000 Berlin 45
Tel.: 0 30 / 831 50 21

Is Gem Da



CAE-Paket CHRON

CHRON: Ein integriertes CAE-Paket

Das **Ingenieurbüro Harald Prillwitz** bietet unter der Bezeichnung **CHRON** ein integriertes CAE-Paket für die Elektronikentwicklung an. Es beinhaltet einen Symboleditor zur einfachen Beschreibung von Bauteilen, einen Schaltplaneditor zur Eingabe einer Schaltung sowie einen Logiksimulator. Das GEM-orientierte Programm erlaubt die einfache Einarbeitung in einen professionellen Logik- und Timingsimulator, der unter anderem eine echte 'Worst Case'-Betrachtung der Schaltung erlaubt.

Die integrierte relationale Datenbank des Systems bietet eine schnelle Schaltungsänderung mit anschließender Neusimulation ohne Zwischenschritte. Schnittstelle zum Layout-System auf ATARI sowie auf anderen Rechnern sind in Vorbereitung. Das Programm, das vollständig mit deutschen Menüs arbeitet, wird mit einem ca. 180-seitigen Handbuch ausgeliefert und ist voraussichtlich ab Februar 1987 zu einem Preis von etwa DM 1.800,– erhältlich.

Harald Prillwitz
Rothplezstr. 23
8000 München 45

ST-Tex

Ein Textverarbeitungsprogramm, das besonders zum Editieren wissenschaftlicher Texte dient, wird von **TOOLS**, einer Softwarefirma aus Bonn angeboten. Das Textformatierprogramm TEX erzeugt Textvorlagen in Buchdruckqualität. Besonders geeignet ist es für mathematische oder andere wissenschaftliche Veröffentlichungen. ST-TEX ist lauffähig auf dem ATARI ST mit mindestens 1 MByte Hauptspeicher und einem doppelseitigen Diskettenlaufwerk. Ein zweites Laufwerk und/oder Harddisk sind zu empfehlen.

$$\int \frac{dx}{ae^{mx} - be^{-mx}} = \begin{cases} \frac{1}{2m\sqrt{ab}} \log \frac{\sqrt{a}e^{mx} - \sqrt{b}}{\sqrt{a}e^{mx} + \sqrt{b}} \\ \frac{1}{m\sqrt{ab}} \tanh^{-1}\left(\frac{\sqrt{a}}{\sqrt{b}}e^{mx}\right) \\ \frac{-1}{m\sqrt{ab}} \coth^{-1}\left(\frac{\sqrt{a}}{\sqrt{b}}e^{mx}\right) \end{cases}$$

Es existiert zur Zeit ein DVI-Treiber, der neben dem Bildschirm in beliebigen Auflösungen verschiedene Drukker unterstützt. Zum Lieferumfang der Grundversion gehört ein Zeichensatz nach Wahl. Der Preis der Grundversion beträgt DM 198,–.

TOOLS GmbH
Kaiserstr. 48
5300 Bonn 1
Tel.: 02 28 / 22 97 91

Endlich lieferbar – OS-9

Das Multiuser/Multitasking-Betriebssystem OS-9, das sich besonders für Echtzeitanwendungen eignet, ist jetzt lieferbar. Zu seinem Lieferumfang gehört das leistungsstarke BASIC 09. Die ATARI-ST-Version ist an die deutsche Tastatur angepaßt und wird über ATARI vertrieben. Das Komplettsystem kostet DM 899,–.

ATARI Corp. (Deutschland) GmbH
Frankfurter Straße 89–91
6096 Raunheim

Desktop Publishing

Unter diesem Zauberwort wird nun das erste Programm in Deutschland angeboten. Das Programm **Publisher** ist eine Textverarbeitung, mit der es möglich ist, fertige Layouts auf dem Bildschirm zu erstellen und auszudrucken. Alle Merkmale moderner Satz- und Layout-Maschinen sind in dieses Programm integriert worden. Mehrspaltiger Seitenumbruch oder getrennte Laufweiten-Einstellung der Buchstaben sind nur einige Merkmale dieses neuen Programmes. Es dürfte interessant für diejenigen sein, die mit Text und Design zu tun haben. Die Ausgabe der Texte kann wahlweise auf Matrix- oder Laser-Druckern oder direkt auf einem Satzbelichter erfolgen. Als Liefertermin wurde uns Februar genannt, leider ist der Preis noch nicht bekannt.

Inzwischen ist uns der neue Preis für das bekannte Zeichenprogramm **Easy Draw** mitgeteilt worden. Die neue Version, die jetzt auch metrische Einheiten verarbeitet, kostet nur noch DM 198,–.

Softline
Schwarzwaldstr. 8 a
7602 Oberkirch

2 Megabyte Speichererweiterung

Wie bekannt, ist der ATARI ST bis maximal 4 MByte aufrüstbar. Die Firma Weide-Elektronik hat, nach alter Manier, eine 2-MByte-Speichererweiterung fertiggestellt. Sie ist mit 16 Chips der neuen 1 Megabit-Speicher ausgerüstet und wird ohne Löten in den Rechner eingebaut. Zu diesen 2 Megabyte Speicherkapazität addieren sich maximal noch 512 KByte des bereits im ATARI vorhandenen Speichers, so daß maximal 2,5 MByte zur Verfügung stehen. Zum Betrieb ist keine Software nötig. Der Verkaufspreis der 2-MByte-Erweiterung beträgt DM 1898,–.

Weide-Elektronik
Regerstr. 34
4010 Hilden

VOC-Royal

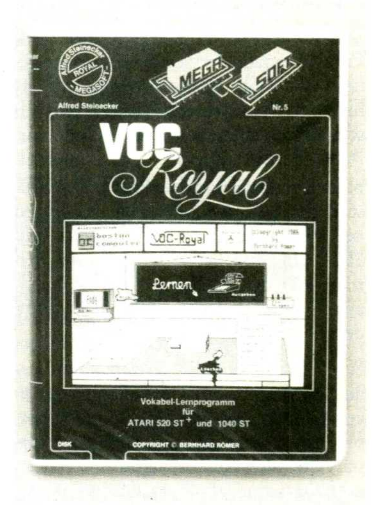
Ein Programm, das beim Erlernen mehrerer Fremdsprachen hilfreich ist, bietet **Boston Computer** aus München. Dieses Vokabellernprogramm wurde für sieben Fremdsprachen entwickelt:

- Englisch
- Französisch
- Italienisch
- Spanisch
- Russisch
- Latein
- Griechisch

Lernpsychologisch liegt dem Programm das Prinzip der Lernkartei zugrunde. Daraus ergibt sich eine Sachlogik, die langatmige Anwendungserklärungen erübrigt. Die neu zu lernenden Wörter oder Redewendungen werden eingegeben. Sie landen im ersten Kasten der Kartei. Beim Lernen wird jeder Begriff abgefragt und wandert, wenn er richtig beantwortet wird, jeweils einen Kasten weiter, bis er nach dem fünften Kasten schließlich ins Lexikon gelangt. Bei einer falschen Antwort fällt der Begriff wieder in den ersten Kasten zurück. Nach Beendigung des Lernvorganges rechnet das Programm den Gesamterfolg aus und stellt diesen als Note und in einer Punktewertung dar.

Das Lexikon kann auch als Nachschlagewerk benutzt werden. Dabei wird der gesuchte Begriff automatisch in den ersten Kasten der Lernkartei zurückgestuft.

Das Programm kostet DM 59,–.



VIDEO-DIGITIZER PRO

Die Weiterentwicklung des Video-Digitizers von Print-Technik ist schon erhältlich. Der Digitizer, der in erster Linie für den professionellen Einsatz in der Industrie gedacht ist, hat eine Auflösung von 512 Pixel in der Horizontalen. In der Vertikalen ist dies von der Anzahl der Bildzeilen des Videosignals abhängig. Die mitgelieferte Software erlaubt das Ablegen, Laden und Darstellen von Bildern. Die Lookup-Tabelle kann abgespeichert und wieder geladen werden. Spiegelung und Invertierung von Bildern ist ebenfalls möglich. Ein Soft-Zoom ermöglicht das Darstellen beliebiger Bildausschnitte. Die Bilder können mit NEOCHROME, DEGAS oder DOODLE weiterverarbeitet werden.

Außerdem wird von Print-Technik ein neuer Videodigitizer vorangekündigt. EXPERT liefert bei Verwendung einer normalen Videokamera 1024 Punkt ein 512 Zeilen. Gleichzeitig digitalisiert er mit einer Genauigkeit von 7 Bit, was einer Anzahl von 128 Graustufen entspricht. Die mitgelieferte Software ermöglicht die Darstellung (sowie die Abspeicherung) von Bildern in allen Auflösungsmodi des ATARI ST. Der Video Digitizer EXPERT wird auf der CEBIT '87 vorgestellt; die Lieferung soll direkt nach der Messe erfolgen.

PRINT-TECHNIK
Nikolaistr. 2
8000 München 40
Tel.: 0 89 / 36 81 97



Digitizerbild von Print Technik

PuzzlePuzzle

Das bekannte Puzzlespiel aus dem Public-Domain-Service bietet, neben seinem Wert als Spiel, die Möglichkeit, hinter die Kulissen der Programmier zu schauen, da diese den Source-Code von PuzzlePuzzle anbieten. Leider traten einige Unklarheiten bezüglich der Kosten des Source-Codes auf. Sie liegen bei insgesamt 35,– DM. Davon sind 15,– DM für den Source Code, den man aber nur erhält, wenn man durch eine Spende von 20,– DM als dankender Benutzer registriert wurde.

Wir hoffen, daß die Fragen damit beantwortet sind. Für Pascal-Programmierer können wir den Source-Code dieses Programms nur empfehlen, da darin viele wertvolle Programmiertricks enthalten sind, die man sich nicht entgehen lassen sollte.

Tommy-Software
Mainzer Landstr. 147
Tel.: 0 69 / 73 69 17

PDNEWS

Auch im Bereich kostenloser Sofware hat sich wieder einiges getan. Das wohl größte Programmwerk ist der Kommandointerpreter von C. Holdack. Er ist den Kommandointerpretern, die in MS-DOS und CP/M Verwendung finden, sehr ähnlich. Die PD-Version bietet fast 30 Befehle und hat neben den üblichen Anweisungen, wie z. B. TYPE, DIR, MD, CD, auch noch einige neue Implemente. Darunter fallen u.a. die Anzeige des freien Speichers, eine Diskinfo und das Ein- und Ausschalten des Cursors. Außerdem ist eine Invertierung des Bildschirms auf Tastendruck möglich. Die Funktionstasten sind frei belegbar, um häufig verwendete Befehle sofort „tastbereit" zu haben.

Auch sonst gibt es einige interessante Programme. Mit IBMDRIVE, einer Spende von Thomas Weinstein, können Daten zwischen IBM- oder IBM-kompatiblen Rechnern und dem ST übertragen werden.

Wer Probleme hatte mit dem Ausdruck von Source-Files oder anderen Texten, bei denen sowohl deutsche als auch amerikanische Sonderzeichen vorkommen, die der Drucker nicht alle zu Papier bringen konnte, dem sei Drupatch von Richard Schmidlechner aus Salzburg empfohlen. Dieses Programm arbeitet mit einer ähnlichen Druckeranpassung wie 1st_word und ist deshalb problemlos an alle Drucker anpassbar. Aufgerufen wird es aus dem Desktop (Desk-Accessoir) und steht deshalb jederzeit zur Verfügung. Durch Beigabe der Source-Codes dient es zusätzlich als Programmierhilfe oder Tip für alle C-Programmierer.

Klaus Mulder schrieb ein Programm zum Umsetzen von ST-Basic- oder Microsoft-Basic-Programmen in GFA-Basic.

Ulrich Viegel vom CIP-Computerclub schrieb ein Lektorprogramm, das einige Fähigkeiten besitzt. Mit ihm kann ein eigenes Lexikon erstellt werden, mit dessen Hilfe Texte korrigiert werden. Die fehlerhaften Wörter können entweder korrigiert, ignoriert oder als richtig in die Lexikon-Datei übernommen werden. Damit vergrößert sich der Wortschatz ständig. Der Text wird in der korrigierten Fassung abgespeichert.

Dem interessierten Künstler stehen die Bilder unseres Kalenderwettbewerbs zur Verfügung, mitsamt einem Programm, das es ermöglicht, Grafik-Bilder mit wesentlich geringerem Platzbedarf auf Diskette abzulegen und sie später wieder in ihre ursprüngliche Größe zurückzuverwandeln.

Die PD-Diskette Nr. 7, auf der sich der XLISP-Interpreter befindet, wurde aktualisiert. Sie enthält nun die Version 1.71, die auch bei unserer Folge 'künstliche Intelligenz' Verwendung findet.

Näheres entnehmen Sie bitte den Public-Domain Seiten.

Modula-2 Update

Vom TDI Modula-2 System ist jetzt eine überarbeitete Version im Vertrieb. Der Compiler bietet zwei neue Optionen, mit denen Sprung-Befehle im Code optimiert werden. Weiterhin sind jetzt der neue Datentyp LONGBITSET und einige kleinere Erweiterungen im Modul SYSTEM enthalten.

Neu sind die Standard-Prozeduren MIN und MAX, die Minima bzw. Maxima zweier Werte ermitteln, sowie PUSH und PUSHADR, mit denen

auf den Systemstack zugegriffen werden kann.

Auch der Modula-Desktop und das Options-Accessory wurden überarbeitet. Insgesamt nimmt das System aufgrund der genannten Optimierung weniger Platz ein.

Den zusätzlich angebotenen Modula-Toolkit gibt es jetzt ebenfalls in einer neuen, von Fehlern befreiten Version.

Bei einem ersten Test des gesamten Systems wurden allerdings einige Fehler gefunden, die schon in der alten Version vorhanden waren.

Für das Frühjahr hat TDI eine völlig neue und erweiterte Version 3.0 angekündigt.

Datenübertragung: ATARI ST als Fotosatz-Erfassungsanlage

Brandneu ist eine Entwicklung der Firma **Fotosatz Mohr**, die es ermöglicht Daten vom ATARI ST in eine Fotosatz-Anlage zu übertragen. Es handelt sich um ein System aus Hard- und Software. Das Programm der Satzanlage, hier Compugraphic MCS-System, wird dazu benutzt die Daten zu verarbeiten, z. B. den Mehrspaltenumbruch herzustellen. Die Kommandos für Schriftgröße, -art, Zeilenvorschub, Satzbreite usw. können sowohl über den ATARI ST als auch direkt über die Tastatur der Satzanlage eingegeben werden.

Interessant ist diese Möglichkeit für alle Druckereien und Verlage welche die bisher unerreichte Qualität der Fotosatzschriften (Randschärfe) nutzen und gleichzeitig die Fehlerquelle durch das nochmalige Eintippen in die Satzanlage vermeiden wollen. Mit dem System wird erreicht, daß die Erfassung von Mengensatz preiswert über den ATARI ST erfolgt. Lediglich für die Überarbeitung wird der Fotosetzer als Spezialist in Anspruch genommen.

Fotosatz Matthias Mohr
Odenwaldstraße 45
6102 Pfungstadt
Tel.: 0 61 57 / 8 29 66

Fußball-Manager für ST

Ein neues spannedes Strategiespiel, das es jedem ermöglicht, in die Rolle eines Fußball-Managers zu schlüpfen, wird für alle Rechner der ATARI ST-Serie angeboten. Dieses Simulationsspiel, das auch nach längerem Spielen seinen Reiz nicht verliert, ist nur für den monochromen Bildschirm gedacht. Als Manager kaufen und verkaufen Sie Spieler, tätigendie saisonalen Ausgaben ihres Vereins, entlassen den Trainer wenn Sie es für richtig halten und beeinflussen durch diese Maßnahmen in vielfältiger Weise den sportlichen und wirtschaftlichen Erfolg oder Mißerfolg Ihres Vereins.

Der Ablauf des Spiels ist dem deutschen Bundesligageschehen nachgeahmt In sportlicher Hinsicht wird um den Gewinn der Meisterschaft bzw. des Pokals gekämpft. Um die Übersichtlichkeit nicht zu verlieren, wird dieses Spiel mit acht Mannschaften gespielt. Das Spiel ist voll GEM-unterstützt und wird mit der Maus und der Tastatur bedient.

Die Vielzahl der Entscheidungen, die Sie treffen, können zum sportlichen und/oder finanziellen Erfolg oder Fiasko führen. Der Spielstand kann jederzeit gespeichert werden.

Dieses anspruchsvolle Spiel wird von **Poffel Products** für DM 49,– angeboten. Zum Lieferumfang gehört ein ausführliches Handbuch. Ein genauer Test dieses Spiels wird in unserem nächsten Heft folgen. Dieses Produkt ist über Händler oder direkt beim Hersteller (siehe unten) zu beziehen.

Poffel Products
Zollhausstraße 39
6520 Worms 26
Tel.: 0 62 41 / 3 49 27

ATARI Mailbox

Ebenfalls zum Thema Datenübertragung erreichte uns quasi in letzter Minute die Nachricht, daß ATARI eine eigene Mailbox eingerichtet hat und diese auch später noch auf Datex erweitern möchte. Die Rufnummer lautet: 0 61 42 / 2 11 61. Folgende Parameter sollten eingestellt werden: 8 Datenbits, keine Parität, 1 Stoppbit, 300 oder 1200 Baud, Vollduplex.

Der ATARI PC

– Nun auch ein MS-DOS Rechner von ATARI

Las Vegas, Nevada, der 8. Januar 1987: Auf der CES-Messe staunten die Anwesenden nicht schlecht, als ATARI-Pressesprecher Neil Harris bekannt gab, daß ATARI einen MS-DOS Kompatiblen auf den Markt bringen wird. Sein wörtlicher Kommentar lautete: „Wir sahen keinen Grund, die Tatsache zu ignorieren, daß wir Gewinne auf dem PC-Kompatiblen-Markt einfahren können. Besonders da dies ein anderer Markt ist, als der, mit dem wir mit unseren Flagschiffen der ST-Serie ansprechen.“

Augenblicklich läuft die PC-Kompatiblen Industrie in zwei Richtungen: Am Ende der einen gibt es die mehr oder weniger anonymen Firmen, die ihre Billig-PC's über die Post verschicken. Die Käufer müssen dann nicht selten feststellen, daß sie noch einige hundert Mark hinzulegen müssen, um auch ein nutzbares System zu besitzen. Am anderen Ende sind dann Firmen wie COMPAQ zu finden, die sogar besser ausgerüstete Maschinen als IBM selbst liefern. Bei Preisen von 1200 Dollar und mehr sind diese Systeme allerdings nur im direkten Vergleich mit IBM zu sehen.

Als Atari anfing, ihren PC zu entwickeln, entschieden sie sich, keinem der beiden Trends zu folgen. Statt dessen will ATARI mit neuer Technologie und dem alten Wahlspruch „Power without the price“ den neuen PC auf den Markt bringen.

Der neue PC soll ca. 500 Dollar kosten und wird alle die Dinge bieten, die sonst nur auf anderen Systemen der höheren Preisklasse zu finden sind. Das Gehäuse mißt 22” x 22” und hat eine Höhe von 2”. Eine 5 1/4-Floppy und Netzteil sind eingebaut. Die abgeteilte Tastatur ähnelt der IBM-XT-Tastatur. Eine zweite 5 1/4- oder auch 3 1/2-Floppy, die in der Lage ist, sowohl ST als auch IBM-Format zu lesen, sind als zusätzliche Option erhältlich.

Der ATARI PC wird mit 512 KByte ausgeliefert und ist wie alle XT's auf 640 KByte auf der Mutterplatine erweiterbar. Als Schnittstellen stehen eine RS232-, eine Parallel, eine Kombinations-Video- und eine ST-ähnliche Schnittstelle zur Verfügung. Ein Mausanschluß, basierend auf dem MICROSOFT INPORT-Chip, wird ebenfalls implementiert sein. Das bedeutet, daß der ATARI PC im Gegensatz zu anderen Mitbewerbern in der Lage sein wird, PC GEM, MICROSOFT Windows und mausorientierte Programme wie MS-Word laufen zu lassen, ohne daß irgendeine Erweiterung nötig wäre.

Der PC besitzt einen 8086-Prozessor, der sowohl mit 4,77 als auch mit 8 MHz getaktet werden kann. Ein Sockel für einen Mathematik-Coprozessor 8087 ist ebenfalls auf der Mutterplatine vorhanden.

Wie man sich denken kann, hat sich ATARI auch im Bereich der Grafik etwas besonderes einfallen lassen. Die meisten Kompatiblen unterstützen nur den IBM-Monochrom-Modus, sind also fast nur zur Textverarbeitung geeignet. Neuerdings kommen die Rechner der gehobenen Preisklasse mit EGA (Enhanced Grafic Adapter)-Grafikkarten auf den Markt. Interessant sind ansonsten noch eine höhere Auflösung im Monochrom-Mode. Dazu diente bisher die sogenannte Hercules-Karte. Doch bei den Grafikkarten gibt es inzwischen eine Unmenge von verschiedenen Versionen verschiedener Hersteller und nicht alle sind mit den IBM-Original-Karten kompatibel.

ATARI hat es jetzt geschafft, die am häufigsten benutzten Karten (IBM-Monochrom, ÇGA, EGA und Hercules) in **einem** Rechner zusammenzufassen. Die ATARI-EGA-Karte ist laut Shiraz Shivji voll abwärts kompatibel.

Ebenfalls passend zum PC hat ATARI einen neuen intelligenten monochromen Monitor angekündigt, der auch Farbbilder erkennt und sie dann in verschiedenen Graustufen darstellen kann. Anhand des ankommenden Videosignals stellt sich der Rechner automatisch auf die entsprechende Grafikkartenemulation um. Der Monitor soll um die 200 Dollar kosten.

Die Frage nach vorhandenen Slots für Zusatzkarten, wird in Zukunft durch eine Erweiterungsbox gelöst werden, da das Gehäuse des ATARI PC kaum mehr Platz dafür bietet.

Der ATARI PC wird auf der CeBit vorgestellt werden und wahrscheinlich ab Mitte des Jahres in Deutschland auf den Markt kommen.

Macintosh Emulatoren auf dem ST

Das Betriebssystem des APPLE Macintosh ist für einen Rechner mit 68000'er Prozessor konzipiert. Die einzelnen Programmteile sind unabhängig von der eigentlichen Hardware des Rechners und nur direkt an den Schnittstellen zum Monitor, zur Diskettenstation oder dem Drucker ist eine Anpassung der Routinen an die Peripherie notwendig. Da der Atari über eine ähnliche Hardware verfügt, und in der Geschwindigkeit des Prozessors und der Auflösung des Monitors besser abschneidet als der Original MAC, ist es sinnvoll, mit dem ST den MAC zu emulieren. Von zwei Firmen werden dafür Emulatoren angeboten: Zum einen von ROBTEK (England) der MAGIC SAC Emulator für 498 DM, und von PROFI-COMP das System ALADIN, das ab Ende Januar für 298 DM erhältlich sein wird.



Bild 1: „Lieferumfang des MAGIC SAC"

MAGIC SAC

Beim MAGIC SAC (Bild 1) wird neben einer Platine mit Platz für die beiden Original MAC ROMS und einer Uhr, noch ein Kabel zum Datenaustausch zwischen MAC und ST geliefert, sowie je eine Diskette für den ST und den MAC. Um den Emulator zum Laufen zu bringen, muß man zuerst das Betriebssystem und die zu benutzenden Programme vom MAC auf den ST übertragen (pro Diskette 12 Minuten) und die ROMs aus dem MAC in die Platine stecken. Vor dem Hochfahren des MAC Betriebssystems läßt der MAGIC SAC noch eine Einstellung der Speichergröße (128 K bis 890 K) zu. Ferner kann der Inhalt einer Diskette in einen Diskcache geladen werden, so daß beim Zugriff auf diese Diskette direkt aus dem Ram geladen wird. Allerdings funktioniert der Diskettencache nicht sehr sicher, so daß man vorher die benutzten Programme noch auf eine andere Diskette speichern sollte.

Obwohl die MAC Oberfläche ähnlich aussieht wie GEM, gibt es doch einige Dinge die anders funktionieren. Zu den Sachen, die man auf jeden Fall sein lassen sollte, gehört das Wechseln von Disketten, ohne dem Betriebssystem

vorher Bescheid zu geben. Das Formatieren von Disketten erfolgt beim MAGIC SAC nur unter GEM, so daß man sich eine ausreichende Menge formatierter Disketten bereithalten sollte. Die Disketten werden ausschließlich einseitig benutzt, so daß bei doppelseitigen Laufwerken einiges an Speicherplatz verschenkt wird. Da das MAC Betriebssystem automatisch auf eine eingelegte Diskette reagiert, und der ST Floppy Controller eine eingelegte Diskette nur bei geschlossenem Schreibschutz erkennen kann, müssen alle Disketten beschreibbar in die Station gelegt werden, damit der MAGIC SAC richtig darauf reagiert. Allerdings ergeben sich bei Disketten, die beim Booten im Laufwerk B: eingelegt waren Schwierigkeiten, da sie nicht erkannt werden.

ALADIN

Das ALADIN System besteht wie der MAGIC SAC aus einer Platine, in die die Original MAC ROMs gesteckt werden. Auf eine zusätzliche Uhr wurde verzichtet, da durch die Benutzung der internen ST Uhr die Kompatibilität zu bestehenden residenten Uhren gewährleistet bleibt. Die Installation bei ALADIN ist um einiges komfortabler und macht auch einen saubereren Eindruck als beim MAGIC SAC. Der MAC Speicher ist frei konfigurierbar bis zu einer Größe von über 900 K Byte. Bei ALADIN ist eine Ramdisk implementiert, die auch beim Reset ihren Inhalt behält, und aus der sogar das System hochgefharen werden kann. Die Disketten können doppelseitig formatiert werden und besitzen dann eine Kapazität von 711 K Byte. Das Formatieren erfolgt bei ALADIN direkt unter dem laufenden Emulator, so daß man nicht erst nach formatierten Disketten suchen muß. Beim Hantieren mit Disketten paßt der Emulator auf, daß ja nicht eine Diskette zum falschen Zeitpunkt aus dem Laufwerk gezogen wird und macht sich sofort durch ein alarmierendes Piepen bemerkbar. Als Erweiterung ist ein Anschluß der ATARI HardDisk als eigenes Laufwerk geplant. Der Drucker kann wie beim MAGIC SAC seriell oder parallel betrieben werden, muß aber zur Zeit noch steuerzeichenkompatibel zum APPLE IMAGEWRITER sein. Als Update ist für ALADIN ein Druckertreiber für EPSON kompatible Drucker in der Entwicklung.



Bild 2: „MS Word“

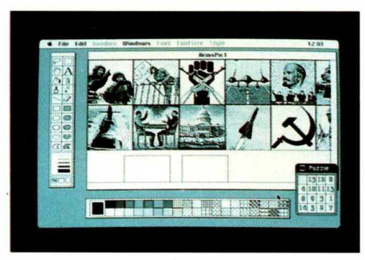
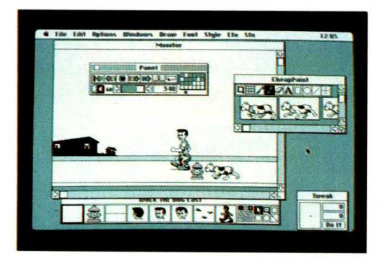

Bild 3: „Full paint“



Bild 4: „Uninvited“


Bild 5: „Video Works“

Eine wichtige Sache beim Emulator ist das richtige Layout der Tastatur: Beim MAGIC SAC ist die amerikanische (englische) Tastaturbelegung des ST eingestellt. Bei ALADIN wird vorrangig die deutsche Tastaturbelegung unterstützt, wahlweise kann aber auch die englische ST Tastatur installiert werden. Zusätzlich werden bei ALADIN die Kodes für den Ziffernblock unterstützt, so daß im Prinzip die neue MAC+ Tastatur vorhanden ist.

Bevor auf einzelne interessante Programme eingegangen wird, sollten einige generelle Aussagen zur Kompatibilität des Emulators gemacht werden. Normalerweise werden auf dem MAC die Programme sehr sauber programmiert, damit sie auch auf den unterschiedlichen MAC Computern laufen. Der Unterschied zwischen einem normalen MAC und zum Beispiel dem MAC XL ist von der Hardware her genauso groß, wie der zwischen MAC und ST, aber trotzdem laufen fast alle Programme auch auf dem XL. Probleme ergeben sich durch die Bytes 0 bis 7, die beim MAC als RAM benutzbar sind, und beim ST nur die Resetadresse des ROMs wiedergeben und nicht veränderbar sind. Ferner kann es passieren, daß Programme, die direkt bestimmte Bereiche im MAC System ROM abfragen, oder direkt auf die Hardware zugreifen, auf dem ST nicht laufen können. Für ALADIN wird ein Programm mitgeliefert werden, das die wichtigen Programme, die direkt mit der Hardware kommunizieren so modifiziert, daß sie auf dem ST laufen können.

Ernsthaftere Komplikationen entstehen bei geschützter MAC Software, da beim Übertragen der Disketten nur die normalen Sektoren übertragen werden können. Aber auch für dieses Problem wird von einigen MAC Softwarefirmen ein sogenanntes HardDisk Utility angeboten, welches die geschützte Software für die Installation auf einer Harddisk vorbereitet, sie also an den Stellen, wo eine Abfrage auf die Originaldiskette erfolgt, entsprechend modifiziert. Diese modifizierten Programme sollten auch auf dem ST ohne Probleme laufen.

Um einen kleinen Einstieg in die Welt der MAC Programme zu geben, erfolgt eine Aufstellung der wichtigsten:

Als erstes Programm das Textprogramm MS WORD (Bild 2), das auf dem ALADIN einwandfrei läuft (MAGIC SAC erliegt einem BUS ERROR). Damit dürfte für professionelle Schreiber ein ausgezeichnetes Textsystem zur Verfügung stehen. Wer mit Grafik zu tun hat kann beim MAGIC SAC mit BILLBOARD (Shareware), einem erweiterten MACPAINT, arbeiten. Bei

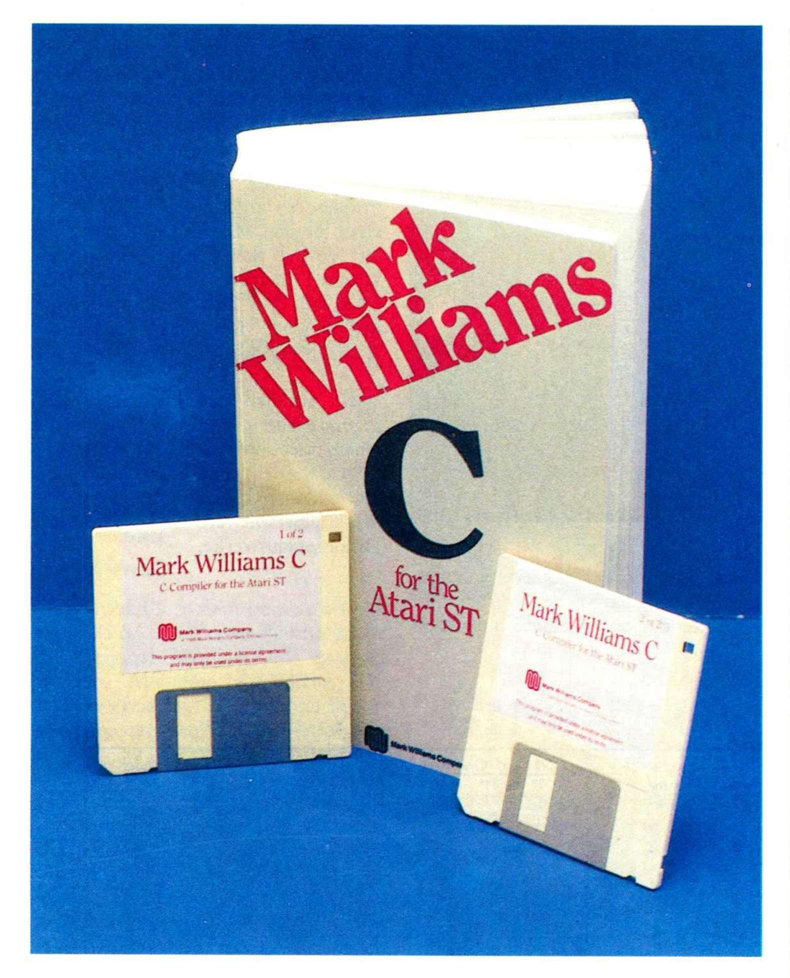


ALADIN empfiehlt sich das Programm FULLPAINT von ANN ARBOR SOFTWORKS (Bild 3), das noch schöner als BILLBOARD ist, aber auf MAGIC SAC nicht läuft. Für Tabellenkalkulation mit integrierter Grafik empfiehlt sich das Microsoft Programm EXCEL, das auf beiden Emulatoren läuft. Wer mit dem Emulator spielen will, kann das mit dem Spiel UNINVITED von MINDSCAPE (Bild 4), wo ein völlig neues Adventurekonzept verwirklicht wurde.

Beim direkten Vergleich der beiden Emulatoren ergibt sich bei der Hardware für den MAGIC SAC ein leichter Vorteil durch eine mitgelieferte Uhr, der allerdings durch einen Mehrpreis von 200 DM mehr als wettgemacht wird. Bei der Software hat ALADIN eindeutig die Nase vorn:

Nicht nur daß wesentlich mehr Programme laufen, und Fehler elegant mit einem adaptierten MAC Debugger behoben werden können, er besitzt auch die eindeutig bessere Implementation der Treibersoftware für Tastatur und Diskettenlaufwerke.

Oliver Joppich



Für MS DOS Rechner ist seit einiger Zeit ein Compiler von Mark Williams verfügbar, der jetzt in einer Umsetzung für den ST erschienen ist. Als Benutzeroberfläche wird eine SHELL benutzt, die sich stark an UNIX anlehnt, so daß für alle UNIX Benutzer eine schnelle Einarbeitungszeit gewährleistet ist. Geliefert wird das System auf 2 doppelseitig bespielten Disketten und einem 643 Seiten starken englischen Handbuch. Die Arbeit mit dem MWC beginnt damit, daß die Programme der Diskette auf eine Harddisk oder eine andere doppelseitige Diskette installiert werden. Das Kopieren der Files wird von einem speziellen INSTALL Programm erledigt. Der ganze Vorgang benötigt einige Zeit (auch bei Harddisk), so daß man erst einmal einen ausführlichen Blick in das Handbuch werfen kann. Es besteht aus 3 Teilen: Einer allgemeinen Beschreibung des Compilersystems (543 Seiten), einer Beschreibung des MAKE Utilities (25 Seiten) und einer abschließenden Erklärung des Editors MicroEMACS, der ein Subset des EMACS Editors darstellt. Der Editor arbeitet konventionell im Text Modus des ST und bietet neben den üblichen Befehlen auch die Aufteilung des Textes in mehrere Abschnitte (im Handbuch Windows genannt) und das Festlegen eines Makros (beliebiger Text kann gespeichert und auf Tastendruck wieder ausgegeben werden). Als besonderer Clou ist der Sourcetext zu diesem Editor mit auf der Diskette (in C), so daß man neue Befehle selbst implementieren kann. Der Compiler-Vorgang kann direkt aus dem Editor gestartet werden, indem eine Kommandozeile direkt an die SHELL übergeben wird. Hier ein kleiner Überblick der Kommandos der mitgelieferten Shell MSH:

mv: Umbenennen eines Files

cp: Kopieren eines Files

rm: File löschen

mkdir/rmdir: Erstellen/Löschen eines Directories

cd: Wechseln des Directories

ls: Zeigt Verzeichnis der Files an (üblich: DIR)

cat/tail: Ausgabe eines Files (üblich: TYPE)

pr: Formatieren eines Textfiles für Druckerausgabe

touch: Datum eines Files aktualisieren

file: Ermittelt den Typ eines Files (Textfile, GEMDOS File, Ordner)

version: Liest bzw. generiert Versionskennzeichen eines Files

cmp/diff: Vergleicht zwei Files auf Unterschiede

size/wc: Gibt Größe / (Anzahl Worte, Zeilen) aus

sort: Sortiert File zeilenweise

uniq: Gibt mehrfach vorhandene Zeilen aus

ar: Startet Archive Programm zum Bearbeiten einer Object-File Library

nm: Gibt die Symbol Tabellen aus Object-Files aus

db: Startet symbolischen Debugger (für MWC C-Files)

crashdmp: Gibt „post-mortum-dump“ nach einem System Absturz aus

drtomw: Konvertiert DIGITAL Objekt Files in MWC Objekt Files

gem: Schaltet Text Modus aus und startet GEM Programm

Die Hauptprogramme in dieser Arbeitsumgebung werden vom Compiler (cc) und Linker (ld) gestellt. Der eigentliche Vorgang des Compilierens geschieht in 4 Durchgängen: Der Preprozessor ersetzt seine Direktiven (★include, ★define, etc.), danach erstellt der Parser den Zwischencode aus dem Source-Text. Dieser wird vom Code-Generator in 68000 Maschinensprache übersetzt. Als letzter Pass wird dieser noch von einem Optimizer abgearbeitet, der auch den Objekt-Code erstellt. Zusätzlich kann noch ein Disassembler zugeschaltet werden, der den Objekt-Code als Assembler File ausgibt. Dieses kann nach kleinen Änderungen mit dem mitgelieferten Assembler wieder in ein Objekt-Format gewandelt werden.

Da die Angabe von Assemblercode im C-File nicht direkt möglich ist, können zeitkritische Routinen nur direkt mit dem Assembler entwickelt werden und werden als Files mit dem Dateityp „S" dem Compiler übergeben.

Die Größe der Datentypen ist beim MWC mit der beim DIGITAL Compiler identisch (also int = 2 Bytes, short = 2 Bytes).

Der Compiler startet normalerweise den Linker automatisch und kann als Eingabe mehrere C-Source-Files und auch Objekt-Files verarbeiten. Auch die Übergabe von Parametern (-Dname = Wert) ist möglich.

Um ein einfaches Programm wie HELLO.C zu compilieren, wird als Befehlszeile das übliche CC HELLO.C eingegeben.

Die Compilation erfolgt dann einwandfrei, aber der Linker meldet, daß in der C-Bibliothek LIBC.A das Verzeichnis der Symbolnamen nicht mehr aktuell ist. Wenn man diese Datei mit dem Archive-Programm bearbeitet, kann das Modul RANLIB.SYM auf den aktuellen Stand gebracht werden. Leider ist diese Fehlermeldung im Handbuch nicht aufgeführt, so daß ich erst nach einiger Zeit die Ursache dafür gefunden hatte.

Die Software umfaßt:

1. Den C-Compiler (cc.ttp, cpp.prg, cc0.prg – cc3.prg)
2. Den Assembler (as.prg)
3. Den Linker (ld.prg)
4. Die Shell, hier MicroShell (msh.prg)
5. Das Makeprogramm (wie von UNIX her bekannt) (make.prg)
6. Den Editor Micro Emacs (me.tos) (zusätzlich als C-Code mitgeliefert)
7. Den Librarian (ar.prg)
8. Den Debugger (db.prg)
9. Das Help-Programm (help.ttp)
10. Weitere 18 Hilfsprogramme, zum größten Teil aus der UNIX/Coherent-Welt stammend.

Programm	Mark Williams C	MEGAMAX	
Dhampstone	17 Sekunden	5 Sekunden	Compilieren
	21 Sekunden	10 Sekunden	Linken
	26 Sekunden	24 Sekunden	Ausführungszeit
	18 931 Bytes	9 765 Bytes	Programmgröße
GEMBIND	9 Sekunden	3 Sekunden	Compilieren
	13 Sekunden	9 Sekunden	Linken
	26 Sekunden	28 Sekunden	Ausführungszeit
	7 493 Bytes	6 861 Bytes	Programmgröße
Hello World	7 Sekunden	2 Sekunden	Compilieren
	13 Sekunden	8 Sekunden	Linken
	7 073 Bytes	6 575 Bytes	Programmgröße

Das Entwanzen von Programmen kann mit dem symbolischen Debugger DB vorgenommen werden, der die lauffähigen Programme auf Assemblerebene mit Angabe der Symbolnamen disassemblieren und ausführen kann.

Um einen Vergleich zu den anderen C-Compilern zu haben, habe ich zum Testen des eigentlichen Compilers die gleichen Programme wie beim MEGAMAX Test im September verwendet. Das MWC System braucht zum Compilieren und Linken der Programme ca. 2 bis 4 mal so lange wie der MEGAMAX. Der erzeugte Kode ist in gewissen Grenzen vergleichbar, allerdings benötigen die Systembibliotheken mehr Speicherplatz.

Zusammenfassend kann gesagt werden, mit dem MWC System erhält man ein sehr umfangreiches C-Entwicklungspaket, mit dem sich Standardprogramme von MSDOS und UNIX einfach auf den ST portieren lassen. Die Unix C-Funktionen, GEMDOS, BIOS und XBIOS sind in der Dokumentation ausführlich beschrieben, allerdings hat man sich die Dokumentation der VDI und AES Funktionen gespart. Die mächtige Shell ist komfortabel gestaltet, arbeitet allerdings nur in einem reinen Textmodus. Wer mit dem MWC vernünftig arbeiten will, sollte eine Harddisk besitzen, da die Systemprogramme mit der Shell nicht auf einer Ramdisk zu installieren sind (es sei denn man hat einen 2 Megabyte Atari) und auf Diskette die Zeiten für das Compilieren und Linken nicht sehr gut sind.

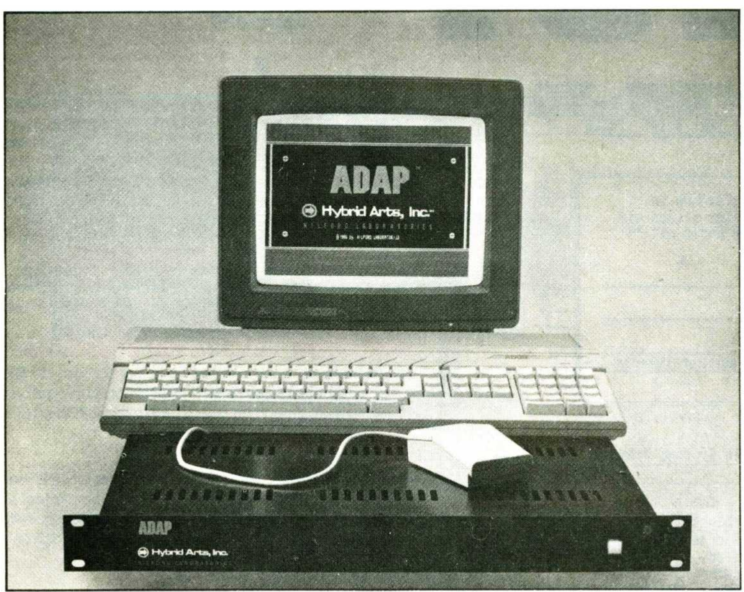
Oliver Joppich

ST 27

Jetzt geht's ab...:

Neue Hard- und Software für Musiker

The ADAP SoundRack™



20 Seconds - 44.1 Khz - 16 bits

Zuerst die Sensation:

Die Firma **Hybrid Arts**, die für ihre ausgezeichnete Midi-Software für die 8-Bit Ataris bekannt ist, hat in Zusammenarbeit mit den **Nilford Laboratories** einen Hardwarezusatz für alle ST-Computer entwickelt, der erstaunliche Dinge leistet. Die Hardware befindet sich in eine 19"-Gehäuse, das mit dem Cartridge Port verbunden wird. Das Gerät nennt sich 'ADAP I', was 'Analog-Digital-Audio-Processor' bedeutet. Die Eins läßt Erweiterungen erwarten, die in der Tat bereits angekündigt wurden und großes (Digital Mastering, d. h. voll digitale Musikaufnahmen) erwarten lassen.

Mit diesem Gerät verwandelt sich der Atari in einen **16-Bit-Sampler**, der mit einer maximalen Samplingrate von **65 (!) kHz stereo** zu sampeln vermag. Bei einer Samplingrate von 41 kHz beträgt die Samplingzeit maximal 20 Sekunden mono, stereo die Hälfte. Wer sich schon einmal mit CD-Spielern beschäftigt hat, weiß, daß 41 kHz Samplingrate bei 16 Bit Auflösung genau den Daten dieser Geräte entspricht. Mit diesem kleinen Zusatzgerät wird also wirklich professionelle Samplingqualität erreicht. Der Preis verspricht ebenfalls sensationell zu werden: Der amerikanische Listenpreis ist 1.995,– Dollar, in Deutschland soll das Gerät um die 5.000,– DM kosten. Zum Vergleich: Das bisher billigste 16 Bit Sampling-System kostet komplett um die 40 000,– DM, es erreicht allerdings auch sehr viel längere Sampling-Zeiten.

Weitere Features des Systems sind:

- Volle Steuerung über Midi, dabei auch Unterstützung des Standard-Midi-Sample-Dump-Format, das es erlaubt, Samples von Geräten mit verschiedenen technischen Daten rein digital auszutauschen.
- Max. 64 Samples gleichzeitig im Speicher.
- Bei 22 kHz Samplingrate 14-stimmig, bei 48 kHz immer noch 6–stimmig polyphon.
- Möglichkeit zur **digitalen** Übernahme von CD-Aufnahmen.
- Auto-Loop Software, findet selbständig optimale Loop-Points im Klangmaterial.
- Oszilloskop-Funktion.
- Erzeugung von Effekten für jedes Sample in **Real-Time**. Jedem einzelnen der maximal 64 Samples, die sich gleichzeitig im Speicher befinden können, können maximal fünf digitale Effekte zugeordnet werden, die in Real-Time beim Erklingen des Tons erzeugt werden, zugeordnet werden. Mitgeliefert werden sollen Hall- und Echoeffekte, weitere können dazugekauft werden. Auf den Klang dieser Effekte darf man gespannt sein, denn die Erzeugung von gutklingendem Hall ist wahrlich ein sehr komplexes, rechenintensives Problem, mit dem der Atari selbst ohne zusätzliche Aufgaben nur mühsam fertig werden dürfte. Man darf also gespannt sein.

Ein ausführlicher Test des 'ADAP'-Soundrack wird folgen, sobald das Gerät lieferbar ist (im Frühjahr '87). Die bisher bekannten Features machen jedenfalls sehr, sehr neugierig.

Midi-Software in rauhen Mengen...

Ebenfalls von **Hybrid Arts**: Eine ganze Reihe von Midi-Programmen wurden für die allernächste Zeit angekündigt. Es handelt sich dabei um einige Sequencer- und Notationsprogramme.

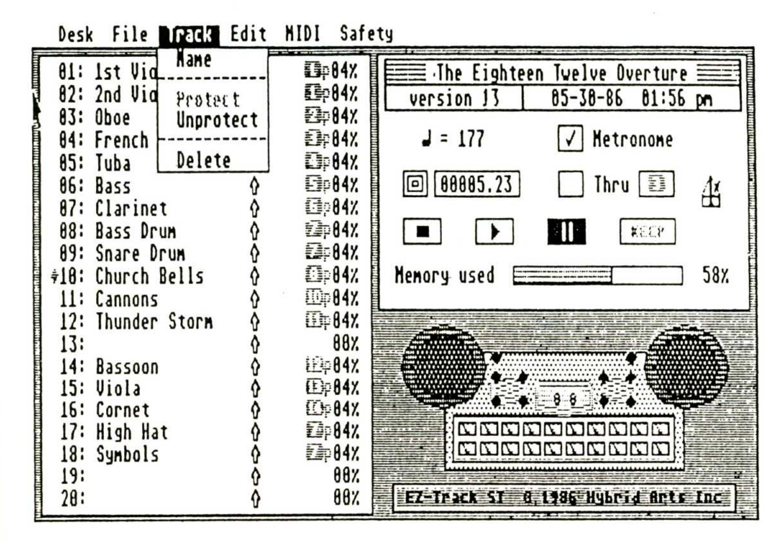


Bild 2: Der einfachste Sequencer von Hybrid Arts

Die Programme bauen in ihren Fähigkeiten aufeinander auf; wer nicht so häufig oder nicht so komplexe Musik erzeugen will, kann einen einfachen 20-Spur Sequencer namens 'EZ-Track ST', dessen Bedienung an eine Tonbandmaschine angelehnt und deshalb sehr einfach ist, haben, wer umfangreiche Editiermöglichkeiten, wie sie ja auch der Steinberg Twenty Four bietet, benötigt, sollte den 60-spurigen 'Miditrack ST' testen, den es auch ('Miditrack ST professional') in einer erweiterten, SMPTE-Timecode kompatiblen Version gibt, die ganz besonders für Filmmusik-Komponisten gedacht ist. Bei dieser Version wird ein Interface, das alle gängigen Timecode- und Synchronisier-Verfahren liest und schreibt, mitgeliefert.

Zu diesen Sequencern gibt es zwei verschiedene Notendruck-Programme, die den verschiedenen Anforderungen angepaßt sind: Das einfache 'EZ-Score ST'-Programm erlaubt die Eingabe von Texten und Akkordsymbolen und ermöglicht die Formatierung der Druckseiten. Das erweiterte 'MidiScore ST Professional' macht zusätzlich auch die Editierung des Notentextes sowie Partiturausdruck möglich. Beide Programme verarbeiten Musik-Files von jedem der drei Hybrid-Arts Sequencer, so daß man sich eine passende Kombination nach Wunsch zusammenstellen kann.

Editor- und Verwaltungsprogramme für verschiedene Midi-Synthesizer werden ebenfalls angeboten. 'GenPatch ST' erlaubt es, Programm-Daten von allen Midi-Instrumenten, die solche Informationen über Midi zur Verfügung stellen, auf Diskette abzuspeichern. Dazu steht ein Auswahlmenü mit den gebräuchlichsten Instrumenten zur Verfügung, mit einem Editor kann das Programm aber auch an andere Instrumente angepaßt werden. Auch ein einfacher Soundeditor für Yamaha's DX-Synthesizer ('DX-Editor ST') steht zur Verfügung.

'DX-Droid' beziehungsweise 'CZ-Droid' nennen sich schließlich Editorprogramme für Yamaha DX- und Casio CZ-Instrumente, die außer den normalen Editor- und Librarie-Funktionen eine sogenannte 'Droid'-Funktion besitzen, die, laut Werbung, mit Hilfe künstlicher Intelligenz eigenständig alte Klänge sinnvoll verändern oder neue Klangpresets für die genannten Instrumente schaffen kann. Mit diesen Werkzeugen dürfte die Erstellung neuer Klänge auf den genannten Synthesizern keine Schwierigkeiten mehr bereiten.

Die Programme sind bis auf 'Midi-Score ST professional', das im Januar zur Verfügung stehen soll, seit Mitte November lieferbar. Den Vertrieb für Deutschland hat die Firma mev-Musik und Elektronik, Postfach 60 01 06 in 8000 München 60 übernommen.

Happy Sampling...

Das 'Sounddesigner'-Programm der Firma Digidesign ist die nächste sehr erfreuliche Ankündigung für alle Benutzer von Sampling-Keyboards wie dem Emulator II, dem Prophet 2000, Ensoniq Mirage, Korg DSS-1 oder ähnlichen. Den Musikern unter MacIntosh-Benutzern ist dieses Programm schon seit längerem bekannt, erlaubt es doch durch seine phänomenalen Nachbearbeitungsmöglichkeiten von digitalisiertem Klangmaterial in Verbindung mit den meisten Samplingkeyboard eine erhebliche Qualitätssteierung, sowie auch mehr Bedienungskomfort. Nun hat Digidesign sich nach langem Zögern doch entschlossen, das Programm auf den Atari umzusetzen. Vermutlich wird es, wie auch bei der MacIntosh-Ausführung spezielle Versionen für die verschiedenen Sampling-Keyboards geben. Einige Features:

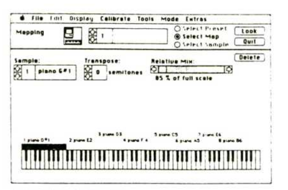
- Die Samples können auf dem Bildschirm dargestellt (Auch mehrere gleichzeitig) und direkt mit einem Zeichenstift manuell bearbeitet werden. Eine dreidimensionale Darstellung (Fourier-Analyse) ist ebenfalls möglich, außerdem auch Ausschneiden, Kopieren, Einkleben und so weiter.
- Eine Loopfinder-Option hilft bei der Suche nach Loop-Points.
- Digitale Equalizer-Funktion.
- Synthese-Option.

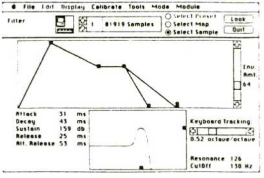
Wenn die Atari-Umsetzung die gleichen Fähigkeiten wie das Original enthält, wird man dieses Programm jedem Sampling-Keyboard-Besitzer uneingeschränkt empfehlen können.

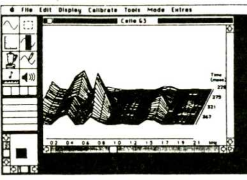
Das Programm wird wahrscheinlich erst zur Musikmesse in Frankfurt lieferbar sein (Februar 87).

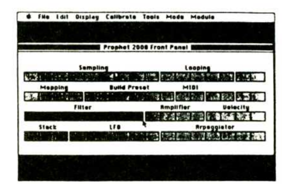
Ein Vertriebsfirma für Digidesign-Programme ist uns bisher unbekannt. Die amerikanische Adresse von Digidesign lautet:

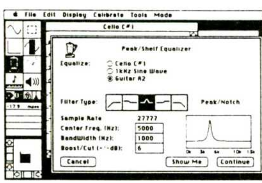
digidesign
920 Commercial Street
Palo Alto CA 94303
USA

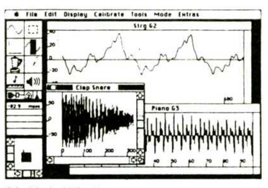

Mapping Module

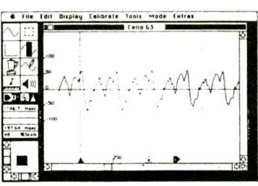

Filter Module


FFT Frequency Analysis


Prophet 2000 Front Panel


Digital Equalization


Multiple Windows


Waveform Display

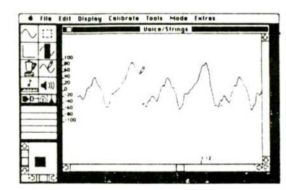

Waveform Drawing

Bild 3: So sieht die MacIntosh-Version des Sounddesigners für den Prophet 2000 aus

Aus deutschen Landen...

Die Kieler Firma **Beam Team** (komisch, die gesamte deutsche Musiksoftware kommt aus dem hohen Norden – Steinberg in Hamburg, Beam Team in Kiel...) hat verschiedene Midi-Programme in Vorbereitung. Bereits lieferbar sind diverse Editoren für Yamaha's DX- und Casio's CZ-Serie. Zur Musikmesse im Februar sollen ein Sequencer samt dazugehörigem Notendruckprogramm erscheinen.

Letzeres soll sich durch eine besonders gute Druckqualität auch mit Matrixdruckern auszeichnen. Warten wir's ab.

Adresse:
Beam Team
Sophienblatt 42a
2300 Kiel 1
Tel.: 04 31 / 6 33 30

Computermischpult für zu Hause...

Die durch ihre Synthesizer-Bausätze bekannte Münchner Firma Doepfer Musikelektronik hat ein Computergesteuertes Mischpult, das als Bausatz oder fertig erhältlich ist, auf den Markt gebracht. Als Steuercomputer kann ein C-64 oder ein Atari ST verwendet werden.

Das Pult besitzt keine Regler mehr (außer für die Eingangsempfindlichkeit), alle Funktionen werden mit Hilfe des Steuercomputers ausgeführt. Es ist modular aufgebaut, kann also den Bedürfnissen des Benutzers sehr gut angepaßt werden.

Zwei Grundausführungen werden angeboten: Eine kleine Ausführung mit 8 Ein- und 4 Ausgängen sowie eine große mit 16 Ein- und 8 Ausgängen. Sowohl zusätzliche Ein- wie Ausgänge können nachträglich angebaut werden. Jeder Eingang läßt sich auf einen oder mehrere Ausgänge legen. Jeder Kanal kann wahlweise mit einem 2- oder 4-fachen parametrischen Equalizer versehen werden.

Die Technischen Daten sind vergleichbar mit normalen Pulten der gleichen Preisklasse.

Die Software ist vollständig GEM-gesteuert und erlaubt sowohl die Speicherung sämtlicher Einstellungen wie die Steuerung über Midi. Für den Live-Einsatz auf der Bühne bedeutet dies, daß man sein gesamtes Instrumentarium inklusive Lautstärke- und Klangeinstellungen, Effekten und Synthesizerpresets mit einem Knopfdruck umstellen kann, ein Vorgang bei dem man mit konventioneller Technik ganz schön ins Schwitzen geraten kann. Im Studio kann ein solches System viel Zeit und damit Geld sparen. Für Programmierer wichtig: Das System ist offen und vollständig dokumentiert, erlaubt also eigene Software-Erweiterungen. Da die Steuerung des eigentlichen Mischpults über Digital-Analog-Wandler erfolgt, kann man die Wandlerkarte auch zur Steuerung eigener Hardware zweckentfremden. Das Doepfer Computermischpult scheint für viele Anwendungen interessant...

Adresse:
Doepfer Musikelektronik
Merianstr. 25
8000 München 19
Tel. 0 89 / 15 64 32

Mal was anderes...

Ausnahmsweise nicht für den ST, aber vielleicht trotzdem interessant: Auf der Tonmeistertagung in München im November stellte Yamaha das erste **volldigitale** Mischpult unter 10 000,– DM vor. Der Prototyp war eine 8-in-2-Ausführung. Das Gerät soll im Mai erscheinen und dann 7.800,– DM kosten. Die technischen Daten dürften in etwa mit guten Analogmischern zu vergleichen sein (Man darf gespannt sein). Dafür enthält der Mischer drei komplette digitale Effektgeräte, deren Qualität dem Multieffektgerät SPX-90 entspricht, wenn auch mit etwas weniger Möglichkeiten. Alle Einstellungen sind abspeicherbar und über Midi kontrollierbar.

Und zu guter letzt...

Das Siegerprogramm unseres Musikwettbewerbs, den Musik-Editor von Eckhardt Kruse, können Sie jetzt auch erwerben (Man muß ihn einfach haben!). Das Programm ist komplett mit deutscher Anleitung beim **Heim-Verlag** erhältlich. Es kostet 44,– DM zuzüglich 5,– DM Porto. Lesen Sie dazu auch den Vergleichstest Musikprogramme in diesem Heft!

Adresse:
Heim Fachverlag
Heidelberger Landstraße 194
6100 Darmstadt 13
Tel. 0 61 51 / 5 60 57

(Christian Schormann)

Gemkurs

Teil 3

von Th. Weinstein

In Teil 3 und 4 meines Kurses „GEM unter C" werde ich recht intensiv in die Objektdatenstrukturen eindringen. Ich will Ihnen zeigen, wie ein Objektbaum aus der Sicht des C-Programmierers aussieht, wie man einen Baum durchlaufen kann, um irgendwelche Operationen auf seine Knoten anzuwenden, und wie man sich eigene Objekte definieren und dann wie Kuckuckseier dem Dialogmanager von AES unterschieben kann. Auch diesmal gibt es wieder Beispiele, die etwas kompakter ausgefallen sind. Dies hat den Vorteil, daß sie vollständig abgedruckt werden können – mit Ausnahme jener Teile, die schon in den ersten beiden Folgen nachzulesen waren.

In der letzten Folge haben Sie gesehen, wie man die Möglichkeiten des Resource Construction Set (RCS) benutzt, um sich schnell einfache Dialoge zu basteln. Diesmal zeige ich Ihnen die darunterliegenden Datenstrukturen und was man mit ihnen machen kann.

Zur Demonstration habe ich einen einfachen Beispieldialog entwickelt, den sie auf Bild 1a sehen. Jedes Objekt im Dialog hat einen Namen bekommen, so daß Sie immer wissen, wovon geredet wird. Normalerweise versieht man nur diejenigen Objekte mit Namen, auf die man im Programm explizit Bezug nehmen will.

Sehen Sie sich bitte parallel zu 1a Bild 1b an. Dort habe ich versucht, die Abhängigkeiten der einzelnen Objekte untereinander zu verdeutlichen. Lassen Sie sich nicht durch die vielen Pfeile verwirren, sondern achten Sie zuerst auf die Anordnung in „Ebenen".

Ganz oben steht ein Objekt mit Namen BEISPIEL. Es ist die sogenannte Wurzel des Objektbaumes. Die Wurzel ist immer dasjenige Objekt im Baum, das mit dem Index 0 bezeichnet wird, sozusagen der Einstieg oder Zugang zu allen anderen Objekten. Bei

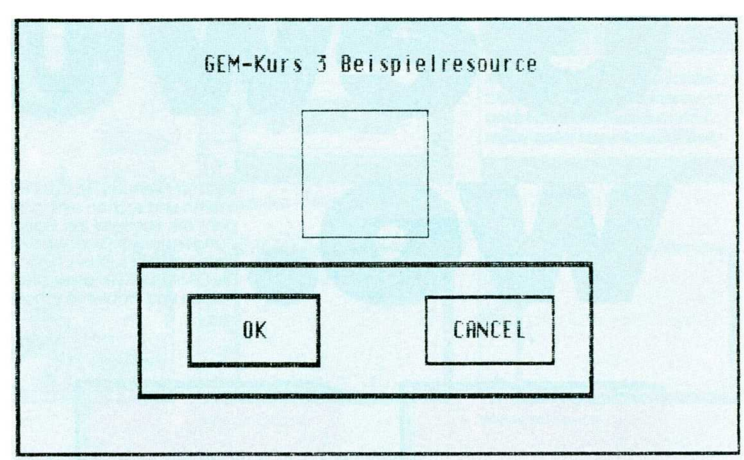


Bild 1a: Beispielresource

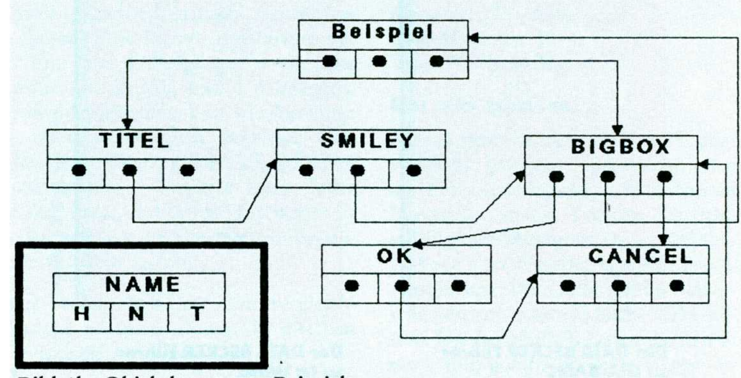


Bild 1b: Objektbaum zur Beispielresource

einem Dialogbaum steht die Wurzel gleichzeitig für den Rahmen um den Dialog. Alle anderen Teile müssen vollständig in diesen Rahmen passen. Das führt uns gleich zu einem weiteren wichtigen Begriff. Zu jedem Objekt gehört eine sogenannte „Bounding Box", also ein umschließendes Rechteck. Die Regeln für GEM-Objektbäume sagen nun, daß jedes im Baum tieferliegende Objekt vollständig innerhalb der Bounding Box seines Vorgängers liegen muß. Außerdem dürfen sich die Bounding Boxes von Objekten auf gleicher Ebene normalerweise nicht überlappen. Aber zurück zum Beispiel: Nach der Wurzel stehen nebeneinander auf gleicher Ebene drei Objekte. Ein String (TITEL), ein Quadrat (SMILEY; Warum ich es so genannt habe, wird erst im nächsten Teil verraten) und ein großes Rechteck mit dickem Rand (BIGBOX). BIGBOX selber hat wiederum zwei untergeordnete Objekte (OK und CANCEL), die üblichen Exit Buttons, die Sie ja schon aus der letzten Folge kennen.

Nachdem die Aufteilung in Ebenen soweit bekannt ist, kommen wir jetzt zu den Pfeilen. Um den Dialog zu zeichnen, muß GEM ja irgendwie von einem Objekt zum Anderen gelangen (möglichst in der richtigen Reihenfolge) und am Ende auch wieder irgendwo definiert aufhören. Dafür werden die Pfeile gebraucht. Sie stellen sozusagen die Wege dar, die den Malalgorithmus von Objekt zu Objekt bringen. Wie Sie sehen, gehen von jedem Objekt maximal drei Wege weiter. Die drei schwarzen Punkte sind jeweils die „Ausgänge" aus dem Objekt. Wenn von einem Objekt kein Weg weiterführt, ist das für GEM ein Zeichen, daß es in einer anderen Richtung weitersuchen muß. Die drei Ausgänge aus einem Objekt haben von links nach rechts die Namen HEAD (Kopf),

```
/************************ BRAIN-WORKS' Resource-Table *********************
 *                                                                        *
 * Dies ist das zur Beispielresource gehörende INCLUDE File nach Sortieren *
 * mit der zum MEGAMAX gehörenden Utility MMTREES.                        *
 *                                                                        *
 **************************************************************************/

#define BEISPIEL   0              /* Baumnr.  0  =  Dialog   */
#define          TITEL      1     /* Objekt      =  STRING   */
#define          SMILEY     2     /* Objekt      =  BOX      */
#define          BIGBOX     3     /* Objekt      =  BOX      */
#define          CANCEL     4     /* Objekt      =  BUTTON   */
#define          OK         5     /* Objekt      =  BUTTON   */

/*************************************************************************
 * Ab hier folgt ein Ausschnitt aus der C Source, die man mit dem RCS    *
 * im ATARI Entwicklungspaket erzeugen kann. Alle Datenstrukturen, die   *
 * in der Beispielresource nicht benutzt werden sind weggelassen.        *
 * Die Kommentare wurden nachträglich von Hand ergänzt.                  *
 *************************************************************************/

/* Hier sind alle in der Resource vorkommenden Strings gesammelt */
char *rs_strings[] = {
"GEM-Kurs 3 Beispielresource",
"CANCEL",
"OK"};

/* Der Kernpunkt: Das Array mit den Objekten */
OBJECT rs_object[] = {
/*
 N   H   T  TYP        FLAGS STATUS    OB_SPEC     X  Y   W   H    NAME
--------------------------------------------------------------------------*/
-1,  1,  3, G_BOX,     NONE, OUTLINED, 0x21100L,   0, 0, 58, 14,  /* BEISPIEL*/
 2, -1, -1, G_STRING,  NONE, NORMAL,   0x0L,      15, 1, 27,  1,  /* TITEL   */
 3, -1, -1, G_BOX,     NONE, NORMAL,   0xFF1100L, 23, 3, 10,  4,  /* SMILEY  */
 0,  4,  5, G_BOX,     NONE, NORMAL,   0xFD1100L, 10, 8, 36,  4,  /* BIGBOX  */
 5, -1, -1, G_BUTTON,  0x5,  NORMAL,   0x1L,      23, 1, 10,  2,  /* CANCEL  */
 3, -1, -1, G_BUTTON,  0x27, NORMAL,   0x2L,       4, 1, 10,  2}; /* OK      */
/*------------------------------------------------------------------------*/

/* Einstieg ins Objektarray. Für jeden Baum einen Eintrag */
long rs_trindex[] = {
0L};
```

```
/*
 * GEMKURS TEIL 3
 *
 * Beispiel fuer das Durchlaufen eines Objektbaumes und das Arbeiten mit
 * der Headerinformation
 *
 * Compiler: MEGAMAX
 */
#include    <define.h>
#include    <stdio.h>
#include    <gemdefs.h>
#include    <obdefs.h>
#include    <osbind.h>

#define  NIL    -1
#define  Wait()    {printf("\033p Weiter - Return \033q"); gemdos(1);}

extern   int    handle;
extern   int    phys_handle;
extern   int    global[];        /* AES Global Array s. Text        */

RSHDR    header;                 /* Resourceheader def in obdefs.h */

main(argc,argv)
int argc;
char **argv;
{
   int        i, ret, hndl, height, room;
   int        *help;
   int        print_object();
   OBJECT  **treeadress,
           *ob;

   setbuf(stdout,0L); /* Ungepufferte Ausgabe */

   appl_init();
   phys_handle = graf_handle(&ret,&ret,&ret,&ret);
   open_vwork();

   /* Einlesen und Ausdrucken des Resourceheaders */
   if ((hndl = Fopen("test.rsc",0)) < 0)
   {
```

NEXT (nächstes) und TAIL (Schwanz). Head führt dabei immer zum ersten (am weitesten links stehenden) Objekt in der eins tieferliegenden Ebene. Tail zeigt immer auf das letzte (ganz rechts stehende) Objekt in der nächsten Ebene. Diese beiden Zeiger klammern also sozusagen den Unterbaum eines Objektes ein. Der Headzeiger von BEISPIEL zeigt, wie in Bild 1b zu sehen ist, auf TITEL, der Tailzeiger auf BIGBOX.

Etwas schwieriger ist es mit dem NEXT-Zeiger. Innerhalb einer Ebene zeigt er auf den rechten Nachfolger des jeweiligen Objekts. Der NEXT-Zeiger des am weitesten rechts stehenden Objekts zeigt eine Stufe höher auf seinen Vater. Auf diese Weise kann man sehr leicht feststellen, daß eine Ebene vollständig abgearbeitet ist. Nur dann nämlich zeigt der TAIL-Zeiger des Objektes, auf das NEXT zeigt, auf das Objekt zurück, dem der NEXT-Zeiger gehört. Das hört sich schwieriger an, als es ist. In Bild 1b z. B. zeigt der NEXT-Zeiger von BIGBOX auf BEISPIEL und dessen TAIL wiederum auf BIGBOX.

Das einzige Objekt in einem Baum, dessen NEXT-Zeiger auf nichts (auf NIL) zeigt, ist die Wurzel (ROOT) des Baumes.

Es gibt verschiedene Möglichkeiten, diese Struktur auf einem Rechner abzubilden. Üblicherweise stellt man Bäume im Speicher des Computers als eine Ansammlung von Knotenobjekten dar, in deren Zeigerkomponenten die ADRESSEN von anderen Objekten im Baum stehen. Digital Research ist bei GEM einen etwas anderen Weg gegangen.

Alle Objekte stehen brav hintereinander in einem großen Array. Die Zeiger der Objekte sind keine Adressen mehr, sondern Indizes, die innerhalb des Arrays auf andere Objekte verweisen. Um diesen Sachverhalt weiter zu erhellen, habe ich die Beispielresource als C Sourcefile ausgedruckt. Das Ergebnis sehen Sie in Listing 1.

Ganz oben sehen Sie das Include File, das der RCS erzeugt, in einer etwas aufbereiteten Form. Es enthält als C Defines die Namen, die ich den Objekten gegeben habe, und dahinter die Indizes, die direkt zum entsprechen-

den Eintrag im Objekt-Array führen. Die Indizes werden dabei von Null aufwärts gezählt. Um also ein bestimmtes Objekt zu finden, schauen Sie unter dem zugehörigen Index nach (für BIGBOX z. B. 3) und zählen dann von 0 ausgehend zeilenweise, bis Sie den Index im Array erreicht haben. Eine Zeile (in Listing 1) im Objektarray steht also immer für ein Objekt.

Schauen wir uns jetzt einen Objekteintrag näher an. In den ersten drei Spalten finden Sie in der Reihenfolge Net, Head und Tail die oben besprochenen Zeiger wieder. Eine −1 steht dabei für leer oder NIL. Den Einstieg in den Baum findet man, ohne die Kommentare anzuschauen, in jener Zeile, in der Next auf −1 steht. Wie oben bemerkt, ist ja die Wurzel das einzige Objekt, auf das dies zutrifft. Versuchen Sie doch selbst einmal, aus dem Objektarray den in Bild 1b gezeichneten Baum zu rekonstruieren.

Die 4. Spalte enthält den Typ des Objekts, im Fall der Wurzel G_BOX. Ausgehend von diesem Typ werden die weiteren Einträge entsprechend interpretiert. Neben den aufgelisteten Typen gibt es noch einige andere, z. B. G_ICON oder G_BOXTEXT.

Eine vollständige Übersicht über alle Typen und die im folgenden noch zu besprechenden Konstanten und Datenstrukturen finden Sie übrigens in obdefs.h, das bei Ihrem C-Compiler dabei sein sollte, falls er GEM unterstützt.

Die fünfte Spalte enthält in Ergänzung zum Typ verschiedene Flags, die durch gesetzte oder nicht gesetzte Bits in einem Integer-Wert symbolisiert werden. In unserem Beispiel können Sie sehen, daß die ersten vier Objekte keine gesetzten Flags haben (NONE). Nur die beiden Buttons (OK, CANCEL) haben Flags gesetzt. Die 0x5 bei CANCEL bedeutet z. B., daß der Button SELECTABLE ist (also auf Anklicken reagiert) und außerdem ein EXIT Button ist. Das heißt der Dialog wird verlassen, wenn er angeklickt wird. Bei OK sind zusätzlich noch die Flags für DEFAULT (er reagiert auf die Return Taste) und LASTOB (er ist das letzte Objekt im Baum) gesetzt.

In der nächsten Spalte steht der augenblickliche Zustand des Objekts. Aus

```
         form_alert(1,"[1][Fehler beim Öffnen von test.rsc][ CRASH ]");
         goto ENDE;
      }
      Fread(hndl,(long) sizeof(RSHDR),&header);
      print_rshdr(&header);
      Fclose(hndl);
      Wait();

      /* Laden der Resource */
      if (!rsrc_load("test.rsc"))
      {
         form_alert(1,"[1][FATAL lieber Thomas.|Keine Resource !][ CRASH ]");
         goto ENDE;
      }

      hide_mouse();
      /* Adresse des Arrays mit den Adressen der Objektbaeume aus dem AES */
      /* GLOBAL Array holen                                                */
      help    = (int *) &treeadress;   /* Trick 17 */
      help[0] = global[5];
      help[1] = global[6];

      /* Informationen ueber Objektbaeume ausgeben */
      for (i = 0; i < header.rsh_ntree; i++)
         treewalk(treeadress[i],ROOT,NIL,print_object);

   ENDE:
      show_mouse();
      close_vwork();
   } /* main() */

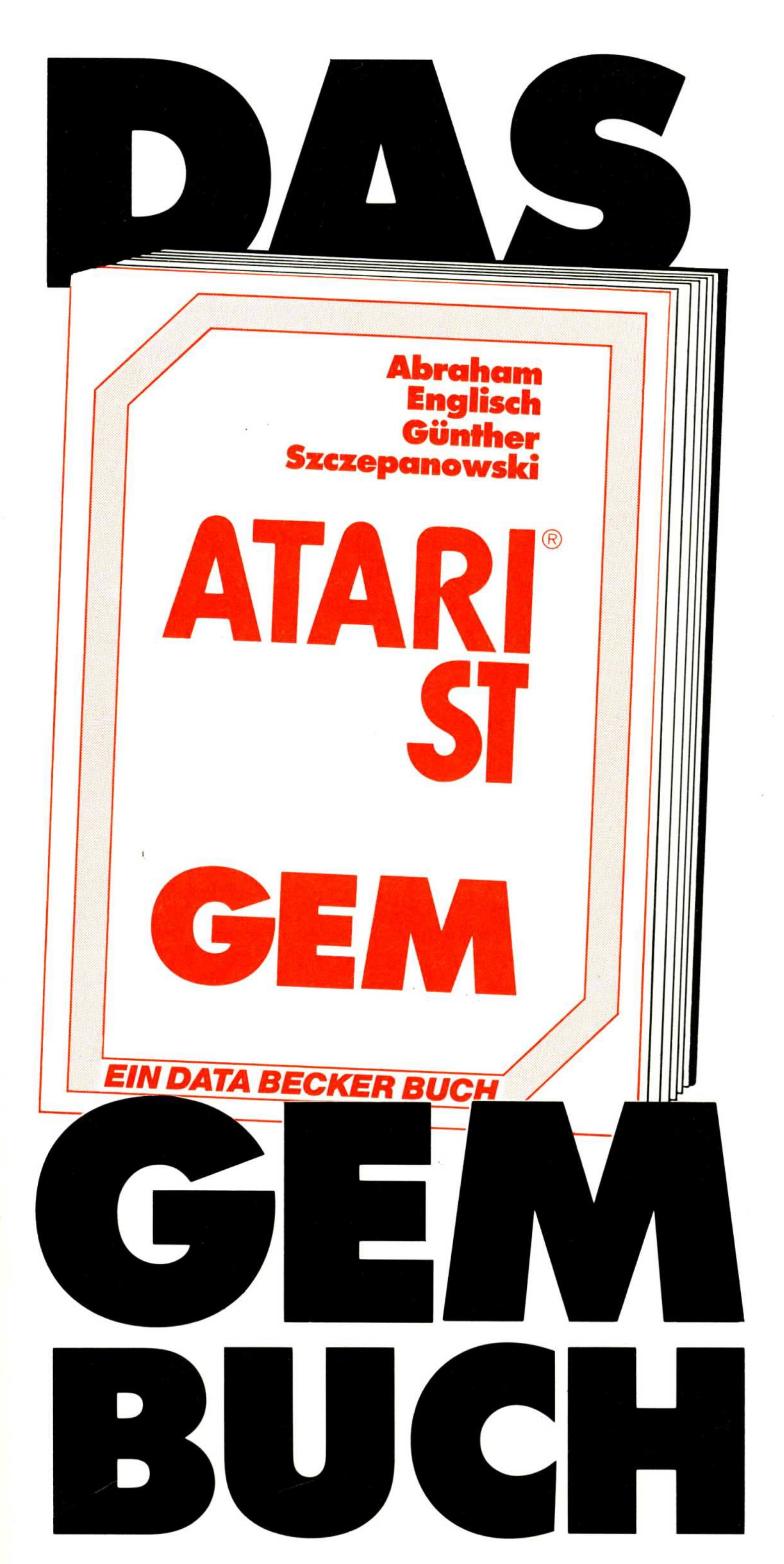
   /*
    * Druckt Informationen zum Objekt index im Baum tree aus und zeichnet
    * das Objekt
    */
   print_object(tree,index)
   OBJECT *tree;
   int     index;
   {
         OBJECT   *o;

         o = tree + index;  /* Objektadresse */

         printf("\033E"); /* CLEAR HOME */
         printf("Adresse des Objekts: 0x%08lx\n",o);
         printf("OB_NEXT   = %d\n",     o->ob_next);
         printf("OB_HEAD   = %d\n",     o->ob_head);
         printf("OB_TAIL   = %d\n",     o->ob_tail);
         printf("OB_TYPE   = %u\n",     o->ob_type);
         printf("OB_FLAGS  = %04x\n",   o->ob_flags);
         printf("OB_STATE  = %04x\n",   o->ob_state);
         printf("OB_SPEC   = 0x%08lx\n",o->ob_spec);
         printf("OB_X      = %d\n",     o->ob_x);
         printf("OB_Y      = %d\n",     o->ob_y);
         printf("OB_WIDTH  = %d\n",     o->ob_width);
         printf("OB_HEIGHT = %d\n\n",   o->ob_height);

         o->ob_x = 639 - o->ob_width;    /* In die rechte untere Ecke legen */
         o->ob_y = 399 - o->ob_height;
         objc_draw(o,0,0,0,0,640,400);   /* Zeichnen und zwar nur das Objekt*/
                                         /* allein.                         */
         Wait(); return(TRUE);           /* Auf Taste warten                */
   }

   /*
    * Resourceheader ausgeben
    */
   print_rshdr(h)
   RSHDR *h;
   {
         printf("\033ERESOURCE HEADER\n\n"); /* CLEAR HOME */
         printf("RSH_VRSN    = %d\n",h->rsh_vrsn);
         printf("RSH_OBJECT  = %d\n",h->rsh_object);
         printf("RSH_TEDINFO = %d\n",h->rsh_tedinfo);
         printf("RSH_ICONBLK = %d\n",h->rsh_iconblk);
         printf("RSH_BITBLK  = %d\n",h->rsh_bitblk);
         printf("RSH_FRSTR   = %d\n",h->rsh_frstr);
         printf("RSH_STRING  = %d\n",h->rsh_string);
         printf("RSH_IMDATA  = %d\n",h->rsh_imdata);
         printf("RSH_FRIMG   = %d\n",h->rsh_frimg);
         printf("RSH_TRINDEX = %d\n",h->rsh_trindex);
         printf("RSH_NOBS    = %d\n",h->rsh_nobs);
         printf("RSH_NTREE   = %d\n",h->rsh_ntree);
         printf("RSH_NTED    = %d\n",h->rsh_nted);
         printf("RSH_NIB     = %d\n",h->rsh_nib);
         printf("RSH_NBB     = %d\n",h->rsh_nbb);
```



Programmieren unter GEM? „Schwierige Einarbeitung und äußerst zeitaufwendig", winkten selbst Insider unlängst ab. Dann kam „Das große GEM-Buch" aus dem Hause DATA BECKER. Ein kompaktes, übersichtlich gegliedertes Buch. Ein Buch voller Lösungen. Es verhalf nicht nur Insidern zum nötigen Durchblick bei der Systemprogrammierung. Und dennoch: Einige wirklich harte Nüsse gab es noch zu knacken. Eine wahre Herausforderung für die Autoren dieses Buches. Sie forschten weiter. Das Ergebnis ihres Ehrgeizes liegt nun vor: ATARI ST GEM. Ein schlichter, präziser Titel für ein Buch, das es in sich hat. Mit einer Fülle an Facts und Informationen zu GEM, wie sie es bisher in ähnlich umfassender Form noch nicht gegeben haben dürfte. Alles, was es zu GEM zu sagen gibt, steht in diesem Buch. Sie erfahren, wie einzelne Funktionen zusammenhängen, nach welchem Konzept GEM aufgebaut ist, was die Systembibliotheken leisten können und vieles mehr. Ist dann genügend Grundwissen vorhanden, legen die Autoren erst richtig los. Systemaufrufe aus GfA-BASIC, C und Assembler, Erstellung eigener GEM-Bindings, Aufbau der Resourcen, Programmierung von Slider-Objekten, Aufbau eines eigenen Desktop und vor allen Dingen: Ein komplett kommentiertes VDI Listing sowie kommentierte Listings ausgewählter AES Funktionen. Alles praxisbezogen und mit vielen Beispielen. Denn ein gut ausgewähltes Listing sagt oft mehr als eine detaillierte Beschreibung. Arbeiten Sie mit diesem Buch, Sie werden vergebens auf die gefürchteten Bömbchen warten. Am Ende liegt Ihnen das gesamte Betriebssystem zu Füßen. Resultat? Nur noch anwenderfreundliche Programme: KLICK.

ATARI ST GEM
Hardcover,
691 Seiten,
DM 69,–

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1

Bitte senden Sie mir:

☐ ATARI ST GEM DM 69,–

zzgl. DM 5,– Versandkosten

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

diesem Eintrag erkennt die Funktion objc_draw, wie sie ein Objekt zu zeichnen hat. Die Dialogbox hat z. B. den Zustand OUTLINED, wird also mit einem dicken Rand gezeichnet. Wie Sie sich vielleicht erinnern, haben wir in der letzten Folge aus dem Statusfeld eines Objektes (ob_state) herausgelesen, welche Objekte der Benutzer während dem Dialog angeklickt hat.

Das Feld ob_spec ist das komplizierteste. Es hat je nach Objekttyp ganz spezielle Funktionen. Bei einer G_BOX z. B. enthält es Informationen über die Breite des Rahmens, sowie Farbe und Muster des Hintergrunds.

Bei den Objekten G_STRING und G_BUTTON enthält es den Index auf ein weiteres Array mit dem Namen rs_strings. Sie finden es in der Mitte von Listing 1. Dort stehen die Texte, die nachher tatsächlich im Objekt auftauchen. Für kompliziertere Objekte wie ICONS oder editierbare Textfelder verweist ob_spec auf weitere Datenstrukturen, die das Objekt näher spezifizieren. Das ob_spec Feld ist, wie wir im nächsten Teil noch sehen werden, der Schlüssel zu selbstdefinierten Objekten.

Die letzten 4 Spalten in einer Objektzeile enthalten die Koordinaten der Bounding Box des Objekts in der Reihenfolge X,Y (linke obere Ecke), W und H (Breite und Höhe). Wie sie leicht sehen können, sind die Werte nicht in Bildschirmpunkten eingetragen (dann müßten sie nämlich viel größer sein). Um ein Resourcefile möglichst geräteunabhängig zu halten, werden sie in Zeichenzeilen und Spalten angegeben; erst die Funktion rsrc_load rechnet sie in Punkte um.

Ein weiterer bemerkenswerter Punkt ist, daß nur die Koordinaten der Wurzel absolut angegeben sind. Alle anderen Objekte sind relativ zum umschließenden Objekt angegeben. Angenehme Folgerung ist, daß man durch Veränderung der Wurzelkoordinaten das gesamte Objekt verschieben kann. Unangenehm ist, daß man, um ein Objekt zu zeichnen, den Baum von der Wurzel aus durchlaufen muß, um die relativen Koordinaten aufzuaddieren (das erledigt jedoch zum Glück objc_draw).

```
        printf("RSH_NSTRING = %d\n",h->rsh_nstring);
        printf("RSH_NIMAGES = %d\n",h->rsh_nimages);
        printf("RSH_RSSIZE  = %d\n\n",h->rsh_rssize);
}

/*
 * TREEWALK Funktion nach Tim Oren
 */
treewalk(tree, first, last, routine)
OBJECT          *tree;          /* Adresse des zu durchlaufenden Baums */
int              first,         /* Index des Startobjekts              */
                 last;          /* Index des letzten Objekts           */
int             (*routine)();   /* Zeiger auf Funktion, die bei jedem  */
                                /* Objekt aufgerufen werden soll       */
{
        register int  this,     /* Index des aktuellen Objekts         */
                      prev;     /* Index des vor 'this' zuletzt be-    */
                                /* suchten Objekts                     */

        prev = this = first;    /* Initialisierung! Tail kann nie auf  */
                                /* sich selbst zeigen !!!              */

        /*
         * Solange weitermachen, bis 'last' Objekt oder Ende des Baums
         */
        while ((this != last) && (this != NIL))
                /*
                 * Wenn wir bei 'this' schon mal waren muss der Tailpointer
                 * von 'this' auf 'prev' zeigen.
                 */
                if (tree[this].ob_tail != prev)
                {
                    prev = this;            /* Ein neuer Knoten! Vorgaenger */
                    this = NIL;             /* neu setzten und 'this' auf   */
                                            /* NIL (Grund s.u.)             */
                    /*
                    * Jetzt fuer den neuen Knoten Funktion aufrufen
                    * Falls die Funktion FALSE liefert wird der gesamte
                    * Unterbaum des Knotens nicht besucht.
                    */
                   if ((*routine)(tree, prev))
                                /*
                                 * Hat der Knoten einen Nachfolger ?
                                 * Sonst ist 'this = NIL.
                                 */
                                this = tree[prev].ob_head;
                    /*
                     * Hat keinen Nachfolger, oder Unterbaum soll
                     * nicht besucht werden.
                     */
                     if (this == NIL)
                     /*
                      * Weiter in der Geschwisterliste des Knotens
                      */
                         this = tree[prev].ob_next;
               }
               else
               /*
                * Sonst ist Unterbaum komplett abgearbeitet und es geht weiter
                * in der Geschwisterliste des Knotens.
                */
               {
                  prev = this;
                  this = tree[prev].ob_next;
               }

} /* End of TREEWALK() */
```

Alle aufgeführten Daten werden also in ein Resourcefile gepackt und müssen dann von der Funktion rsrc_load entsprechend interpretiert werden.

Damit Sie ein Gefühl für Resourcen bekommen und verschiedene Aspekte selbst nachvollziehen können, habe ich ein kleines Beispielprogramm geschrieben. Sie finden es in Listing 2. Sie können sich damit beliebige Resourcefiles – Ihre eigenen oder die von anderen Programmen – schrittweise anschauen. Es ist vollständig abgedruckt, bis auf die vier Funktionen open_vwork, close_work, hide_mouse und show_mouse, die schon in Teil 1 des Kurses erklärt wurden.

Als nächstes werde ich die interessanten Teile daraus erklären. Zuerst ein paar Worte zur Bedienung. Das Programm liest jede beliebige Resourcedatei ein. Einzige Bedingung ist, daß sie den Namen TEST.RSC hat. Zuerst müssen Sie also die Resourcedatei, die Sie anschauen wollen, umbenennen.

Das Programm zeigt Ihnen dann, was im Header der Datei steht, und druckt anschließend für jeden Baum und jedes Objekt in der Reihenfolge, in der sie gezeichnet werden, die Objektdatenstruktur aus, und zeichnet das Objekt außerdem noch in der rechten unteren Bildschirmecke. Schauen Sie es sich einfach an. Es heißt RSC-LOOK und steht wie üblich inklusive Source und allem anderen auf der vom Verlag erhältlichen Monats-Diskette.

Nach der üblichen Initialisierung am Anfang wird in den Zeilen 40–49 zuerst der Header der Resourcedatei gelesen und ausgegeben. Neben dem angenehmen Effekt, daß man sich die Daten anschauen kann, hat das folgenden Grund: Da jede beliebige Resourcedatei geladen werden kann, muß man sich irgendwo die Information besorgen, wieviele Bäume überhaupt vorhanden sind. Genau diese Information findet sich im Resourceheader. Fällt Ihnen noch eine andere Methode ein? Aus der Erklärung der Verzeigerung weiter oben kann man eine alternative Methode ableiten.

Dann wird die Resource ganz normal mit rsrc__load geladen. Das nimmt uns die Arbeit ab, die Verzeigerung selbst einrichten zu müssen. Jetzt folgt ein weiterer Trick. Jede AES-Applikation hat ein globales Array GLOBAL, in dem allgemeine Informationen zur Applikation eingetragen werden. Unter anderem steht in GLOBAL[5] und GLOABL[6] die Adresse eines Arrays, in dem die Adressen der Wurzeln sämtlicher Bäume stehen. Mit dieser Information und der Anzahl aus dem Header kann man die Bäume jetzt nacheinander durchlaufen.

Damit stoßen wir zum Kernpunkt des Programms vor: Dem Durchlaufen eines Baumes. Ich muß an dieser Stelle zugeben, daß dieser Programmteil nicht allein von mir stammt. Ich habe ihn aus einem Artikel von Tim Oren, dem Autor des DR RCS entnommen, ausführlicher kommentiert und die Variablen etwas sinniger benannt. Mir war zuerst die Idee zu einem rekursiven Algorithmus gekommen, der zwar wunderbar funktionierte, doch ebenso umfangreich wie schwer verständlich war.

Die Funktion von Tim Oren ist ziemlich allgemein angelegt und kann immer dann benutzt werden, wenn ein Objektebaum oder Teile davon, Knoten für Knoten, bearbeitet werden müssen.

Für die weiteren Erklärungen sollten Sie sich Listing 2 ab Zeile 135 vornehmen.

Die Funktion treewalk wird mit drei Parametern aufgerufen. Zuerst wird die Adresse des zu übertragenden Baums übergeben, dann der Index des ersten zu bearbeitenden Objekts (meistens wird das das Objekt ROOT also die Wurzel sein). Der dritte Parameter steht für den Index des letzten zu bearbeitenden Objekts (wenn hier NIL übergeben wird, wird der gesamte Baum abgearbeitet). An vierter Stelle wird die Adresse einer Funktion übergeben, die jedesmal dann aufgerufen wird, wenn ein neuer Knoten im Baum erreicht wird. Diese Funktion hat den Nebeneffekt, daß, wenn sie als Ergebnis FALSE liefert, der gesamte Unterbaum, also der Baum, auf den die HEAD Komponente des Objekts zeigt, nicht bearbeitet wird.

Falls Sie bis jetzt nicht wußten, wofür es in C Funktionszeiger gibt – hier sehen Sie ein Beispiel. In Pascal müßte man sich für jeden Anwendungsfall eine eigene Prozedur schreiben.

Das Interessante an der Funktion ist die Tatsache, daß sie vollkommen iterativ (schrittweise), also ohne Rekursion arbeitet und trotzdem sehr elegant – in programmtechnischem Sinne – ist.

Da sich die Funktion mit Worten nur schlecht erklären läßt, werde ich nur kurz die Idee skizzieren. Sie sind dann aufgefordert, sich mit einem Bleistift und einem Blatt Papier zu bewaffnen und C-Interpreter zu spielen; vielleicht mit Hilfe des Beispielbaums in Bild 1b.

Solange das aktuelle Objekt (this) ungleich dem Letzten (last) und ungleich NIL (–1) ist, wird die while-Schleife durchlaufen.

Der eigentliche Trick an der Sache ist es, immer den Index des Letzten (prev) vor dem aktuellen behandelten Objektes mitzuziehen und damit festzustellen, wann eine Hierarchieebene abgearbeitet ist. Sie wissen ja noch: Wenn der Tailzeiger des aktuellen Objekts auf das zuletzt bearbeitete Objekt zeigt, ist man eine Stufe im Baum aufgestiegen. Ansonsten versucht der Algorithmus, sich immer möglichst tief in den Baum zu schaffen, indem er solange als möglich dem Pfad folgt, der durch den HEAD-Zeiger vorgegeben ist. Erst wenn es in dieser Richtung nicht mehr weitergeht, wird über NEXT ein Schritt in „horizontaler" Richtung gemacht. Der Fachausdruck, für diese Art einen Baum zu durchlaufen, heißt Tiefensuche oder Depth First oder auch „den Baum in Preorder" durchlaufen.

treewalk wird in Listing 2 verwendet, um für jeden Knoten die Funktion print__object aufzurufen. An print__object ist nur die Art und Weise bemerkenswert, wie draw__objc benutzt wird, um genau ein Objekt zu zeichnen.

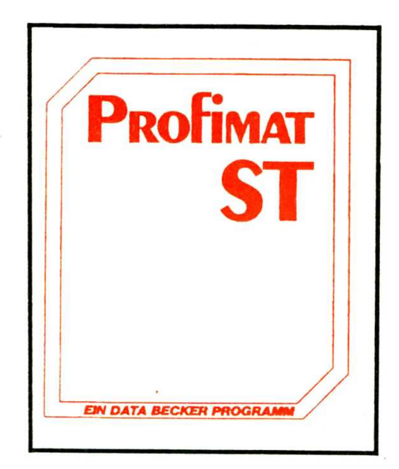
Spielen Sie etwas mit dem Programm und schauen Sie sich mal die Resourcen zu verschiedenen Programmen an. Interessant ist z. B. die Art und Weise, in der Pulldown-Menüs gemacht werden.

So, ich denke, es ist mal wieder genug. Sie wissen jetzt so ziemlich alles über den globalen Aufbau von Objektbäumen. Wie die Objekte im Einzelnen aussehen (Icons, Textfelder, Buttons), wurde schon an verschiedenen Stellen in aller Ausführlichkeit abgedruckt.

In der nächsten Folge zeige ich Ihnen, wie man beliebige eigene Objekte in die Resourcen einbauen kann und wie man mit dem TOUCHEXIT-Attribut den **form__do** Aufruf leicht erweitern kann.

Der Profimat ST von Data Becker

Wie professionell ist dieser Makroassembler?




```
Desk  Datei  Assembler  Debugger  Editor  Suchen  Block  Tabelle
                              Assembler
Frei : 198618 , TEXT :       0 , DATA :       0 , BSS :       0 , Rest : 177618
                              Editor
Z1:  202, Sp:   13, Top:  191, Ges.:  872, Einf ,Frei:  59677, Rest: 177
decode6:    move.b   #'b',(text_pos)+  ; 'b' in den Textpuffer
            move.w   (prg_zeiger),daten  ; 1. Prg.-Datenwort -> D0
            andi.w   #$f00,daten       ; Bedingung heraussuchen
            cmp.w    #$200,daten       ; auf Bedingung groesser 1 test
            bpl      norm_bed          ; Normale Bedingung
            lsr.w    #7,daten          ; Bedingung wird zum Index
            lea      endung,adresse    ; Zeiger auf die Texte
            move.b   0(adresse,daten),(text_pos)+  ; 1. Zeichen in den
            move.b   1(adresse,daten),(text_pos)+  ; 2. Zeichen ebenfa
            bsr      z_daten           ; Textzeiger auf Datenbereich
            bra      kein_bed          ; Kein Bedingungscode
;
norm_bed:   bsr      bed_aus           ; Bedingung in den Puffer einfu
kein_bed:   move.w   (prg_zeiger),daten  ; 8 Bit Adressdistanzwert ho
            andi.l   #$ff,daten        ; Unteres Byte
```

Bild 1: Alles in einem: Profimat ST

Mit dem Profimat ST demonstriert Data Becker ein neues Assemblerkonzept. Was es bisher nur unter TOS gab, wird jetzt mit GEM-Unterstützung geboten: Ein integriertes Paket bestehend aus Editor, Makroassembler, Reassembler und Debugger.

Alle Bestandteile des Programmes befinden sich simultan im Speicher. Man kann also blitzschnell vom Assembler- in das Editor- oder das Debuggerfenster wechseln. Dadurch gestattet Profimat sehr schnelle Programmentwicklung. Das ewige Laden der einzelnen Programmteile und wiederholte Abspeichern der Programmquelltexte bei der Fehlerkorrektur entfällt.

Weiterhin ist im Lieferumfang ein 170 Seiten starkes Anleitungsbuch enthalten, in dem alle Funktionen des Programmes mit Beispielen erläutert werden. Außerdem wird eine sehr umfangreiche Makrobibliothek mitgeliefert, die alle Betriebssystemaufrufe samt denen des BIOS, XBIOS und von GEM enthält. Für die ersten Programmierversuche findet man auf der Diskette zusätzlich diverse Beispielprogramme, so daß man nicht lange braucht, um selbst Programme mit Profimat zu erstellen.

Zuerst wird editiert...

Mit dem Profimat-Editor läßt es sich bereits nach einer kurzen Einarbeitungszeit effektiv arbeiten. Er orientiert sich in der Bedienung an 1st_Word, ist aber außer dem Scrollen mit dem Cursor um einiges schneller. Außerdem bietet er einige nützliche Sonderfunktionen. Seine Funktionstasten sind frei belegbar und durch eigene Makros auf den Funktionstasten können Sie sich oft die Arbeit vereinfachen. Haben Sie einmal vergessen, wie die Funktionstasten belegt sind, so genügt ein Knopfdruck und diese wird in der Infozeile angezeigt. In dieser Infozeile lassen sich auch längere Befehle an den Editor eingeben. Im Normalfall werden dort allerdings die Cursorposition und weitere Daten dargestellt (Bild 1).

Der Editor arbeitet mit dem Assembler zusammen und kann zur Fehlerkorrektur die von diesem erzeugbare Fehlerdatei übernehmen.

...und dann assembliert.

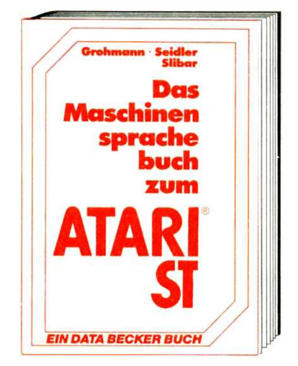
Zweifellos der wichtigste Bestandteil ist der Assembler. Er kann wie schon erwähnt auch Makros verarbeiten. Makros sind Gruppen von Assemblerbefehlen, die einen Namen tragen. Ein Auftreten dieses Namens im Quelltext veranlaßt den Assembler hier die komplette Befehlsgruppe – eventuell auch angepaßt – einzusetzen. Außer der Makrofähigkeit unterstützt Profimat noch bedingte und wiederholte Assemblierung. Mit seinen mehr als 20 Pseudoopcodes lassen sich die meisten Programmieraufgaben lösen.

Nun zur Geschwindigkeit des Assemblers. Zur Überprüfung derselben wurde eine ca. 45 KByte langer kommentierter Quelltext ohne Makros assembliert. Das gleiche wurde mit demsel-

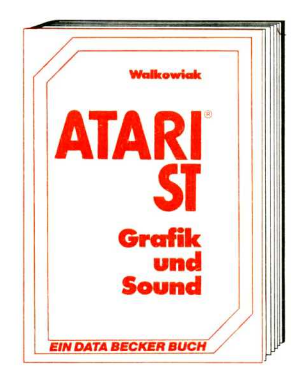
BESTSELLER



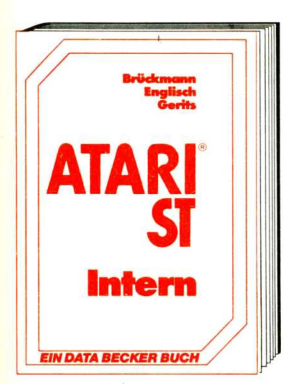
Was passiert, wenn ein BASIC-Interpreter neue Maßstäbe setzt? Dann sucht DATA BECKER dazu einen Buchautor, der ebenfalls neue Maßstäbe setzt. Das Ergebnis ist ein Buch, in dem Ihnen anhand eines Programmes alle GfA-BASIC-Befehle erläutert werden – und zwar immer genau dann, wenn Sie sie brauchen. So entsteht das Graphic-Construction-Set. Ein Grafik-Programm der Superlative. Dieses Buch macht endlich Schluß mit nackten Befehlsübersichten.
Das große GfA-BASIC-Buch
Hardcover, 486 Seiten, DM 49,–



68000-Assembler – der Schlüssel zur Programmierung der spektakulärsten Effekte und der leistungsfähigsten Programme auf dem ATARI ST. Dieses Buch zeigt Ihnen, wie Sie Ihren ST mit Maschinensprache voll ausnutzen können: Verwendung von Systemroutinen, Bitmanipulationen, Rekursion, Stacks, Einbinden von Assemblerroutinen in Hochsprachen – vom ersten Schritt bis hin zum echten Einsatz von Maschinensprache: dieses Buch begleitet Sie.
ATARI ST Maschinensprache
334 Seiten, DM 39,–



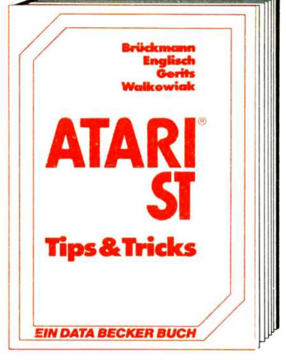
Hören und sehen Sie, was Ihr ST leisten kann – mit ATARI ST Grafik und Sound. Arbeiten Sie mit Raster- und Vektorgrafik, programmieren Sie einen 2- und 3-D-Funktionsplotter, erfahren Sie alles zum Thema Computeranimation auf dem ST, lernen Sie mehr über die Grundlagen der synthetischen Musik... All das wird ausführlich mit vielen Beispielen erklärt. Entdecken Sie die Welt der Grafik- und Soundprogrammierung.
ATARI ST Grafik & Sound
296 Seiten, DM 49,–



Das Informationspaket zum ATARI ST. Geschnürt vom bewährten INTERN-Team Gerits, Englisch, Brückmann. Ein paar Stichworte aus dem Inhalt: Der 68000-Prozessor, der I/O-Controller MFP 68901, der Soundchip YM-2149, alles über die Schnittstellen des ST und XBIOS, kommentiertes BIOS-Listing, wichtige Systemadressen, MIDI-Interface, Grafikaufbau – dieser kleine Ausschnitt reicht sicher, um klarzumachen: Das ist Pflichtlektüre für ST-Profis.
ATARI ST Intern
Hardcover, 506 Seiten, DM 69,–



Endlich eine umfassende Analyse zu Floppy und Harddisk des ATARI ST. Ein Buch, das sogar die Profis noch erstaunen wird – aber trotzdem auch für den Einsteiger geeignet ist. Dafür sorgen viele ausführliche Beispiele. Controllerbeschreibung, Programmierung unter TOS, BIOS und XBIOS, Boot-Sektor, Programmformate auf Diskette und Festplatte, RAM-Disk – in diesem Buch findet sich alles, was dazugehört.
ATARI ST Floppy und Harddisk
Hardcover, 522 Seiten, DM 59,–



Randvoll mit guten Ideen: Das ist unser Tips und Tricks zum ATARI ST. Sie suchen einen Druckerspooler? In diesem Buch finden Sie ihn. Sie brauchen eine Farb-Hardcopy-Routine? Bitteschön. Eine RAM-Disk anlegen? Nichts leichter als das. Viele neue Routinen, die problemlos in eigene Programme eingebunden werden können, warten nur darauf, daß Sie sie ausprobieren.
ATARI ST Tips & Tricks
Hardcover,
352 Seiten
DM 49,–

NEU: ST-INFO

Mehr über unser großes Angebot aktueller Bücher und Programme zum ATARI ST enthält unser neues ST-INFO, das wir Ihnen gerne zusenden.

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Bild 2: Der übersichtliche Debugger

Bild 3: Übersichtliche Tabellen helfen bei der Arbeit

ben Text gemacht, nachdem dessen Kommentare entfernt wurden:

	Diskette	im Speicher
Mit Kommentar (1180 Zeilen / 45 KByte Quelltext)	86 s	25 s
Ohne Kommentar (780 Zeilen / 19 KByte Quelltext)	50 s	18 s

Daraus ergibt sich ein Durchsatz von ca. 15 Zeilen/s bei Betrieb mit dem Diskettenlaufwerk und ca. 45 Zeilen/s bei Assemblierung im Speicher (bei Profimat wählbar).

Gefehlt hat mir an diesem Assembler nur eine Möglichkeit, Objektdateien zu erstellen, die sich mit einem Linker in andere Programme höherer Sprachen wie 'C' oder Pascal einbinden lassen. Es wird lediglich gestattet, Binärdateien, d. h. zum Beispiel in Basic ladbare, zu erzeugen.

Es geht auch umgekehrt:

Wenn es Ihnen über ist, selbst Programme zu schreiben oder Ihnen zu irgendeinem Programm die Quelltexte fehlen, so können Sie Profimat ST veranlassen, aus einer Programmdatei einen Quelltext zu generieren. Dies kann sowohl auf dem Drucker als auch auf Diskette geschehen. Der eingebaute Reassembler ersetzt automatisch Adressen durch Labels und Konstanten im Programm durch Symbole. Sie erhalten so ein wieder assemblierbares Programm.

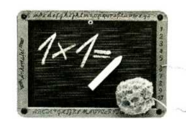
Der Debugger

Den Debugger des Profimat kann man ohne weiteres als gelungen bezeichnen. Durch die Nutzung des gesamten Bildschirms erscheinen alle relevanten Daten übersichtlich gegliedert (Bild 2). Im rechten Bereich kann entweder der Speicherinhalt editiert oder disassembliert angezeigt werden. Mit der Maus lassen sich dort Breakpoints setzen und verschieben. Zu testende Programme werden an diesen Breakpoints in ihrer Ausführung unterbrochen und eine Rückkehr in den Debugger veranlaßt. Im mittleren und unteren Teil kann man verschiedene Parameter des Debuggers einstellen und Befehle aufrufen. Dort läßt sich unter anderem auch ein 68020-Emulationsmodus aufrufen. Der 68020 ist der größere Bruder des 68000, ein reiner 32-Bit-Prozessor. Alle Befehle dieses Chips werden, falls der Emulationsmodus angewählt wurde von Profimat simuliert. Schließlich kann man sich im linken Bildschirmbereich noch verschiedene Register- oder Speicherzelleninhalte anzeigen lassen – welche dies sind, ist frei wählbar.

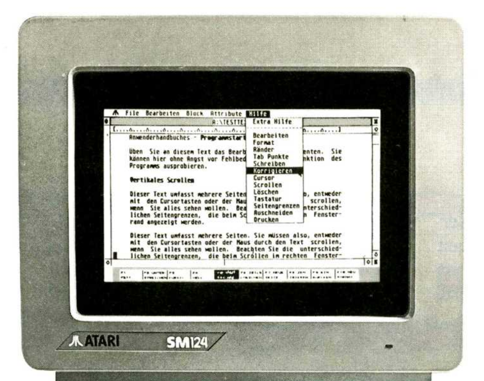
Zum Nachschlagen

Ein praktisches Detail von Profimat findet man unter dem Menüeintrag „Tabelle“. Dort kannman aus allen Programmteilen heraus zwischen effektiven Adressen, GEMDOS, BIOS, XBIOS, VDI und AES eine Tabelle auswählen und darin blättern. Alle diese Tabellen werden ständig im Speicher gehalten und stehen so jederzeit zur Verfügung. Sollten Sie beim Editieren vergessen haben, welche effektiven Adressen z. B. der Move-Befehl erlaubt, so schauen Sie einfach nach (Bild 3). Vor allem bei den vielen Betriebssystemroutinen ist es nützlich, wenn man mal schnell nachsehen kann, welche Parameter der eine oder andere Befehl benötigt. Interessanterweise kann man die Erklärungen auch editieren und den Menüpunkt durch weitere Tabellen ergänzen.

Einer für Alle(s):



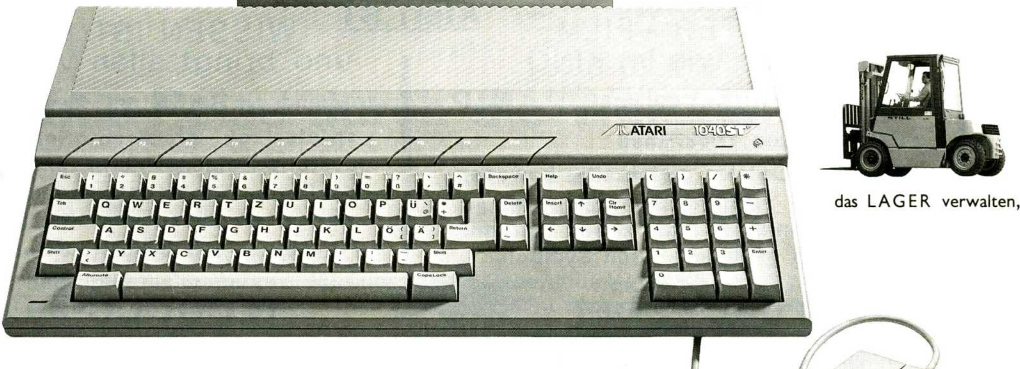
Ob Sie an der UNI pauken oder lehren,



in Ihrer PRAXIS auf den Zahn fühlen,



im LABOR versuchen,



das LAGER verwalten,



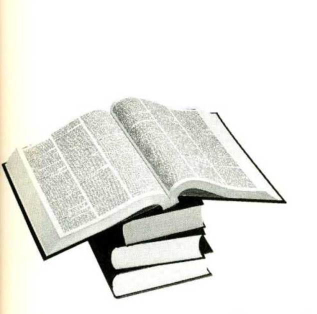
mit MUSIK duren und mollen,



in Ihrer KNEIPE die Wirtschaft ankurbeln,



als ENTSCHEIDUNGSTRÄGER Zeichen setzen,





Unmengen an TEXTen verarbeiten,



große und kleine Kinder VERSICHERN . . .

Die ATARI ST-Personalcomputer haben alle Vorteile, die Sie von leistungsstarken 16/32-bit Computern erwarten können. Sei es die Speicherkapazität, die hohe Arbeitsgeschwindigkeit, die bestechende Grafik, die Schnittstellen . . .

In dieser Leistungsklasse hat ATARI die Maßstäbe gesetzt. Auch beim Preis! Computerleistung zu solch niedrigen Preisen kann nur anbieten, wer modernste Technologien einsetzt.

ATARI ST, das ist Computertechnologie für Menschen, die mehr leisten wollen.

Und. Jede Menge Software-Programme gibt es für die ATARI ST. Welche Anforderungen Sie auch stellen, welche Aufgaben Sie auch lösen wollen.

ATARI ST-Personalcomputer gibt es beim Fachhandel und in den Fachabteilungen der Warenhäuser.



. . . wir machen Spitzentechnologie preiswert.

Der ATARI ST ist einer für Alle(s).

Abschließendes

Das Programm ist mit einem Kopierschutz versehen, der bei der Arbeit nicht weiter stört, nur daß beim Start des Programmes die Originaldiskette eingelegt werden muß. Wer Angst um seine Diskette hat, kann bei Data Becker für DM 20,– eine Sicherheitskopie bestellen. Das Programm selbst, das mit seinen Leistungsmerkmalen sowohl für den Anfänger als auch Fortgeschrittenen geeignet ist, wird für den gebotenen Leistungsumfang zum äußerst fairen Preis von DM 99,– vertrieben. (Jörg Wilhelm)

PROLOG 10

Gratis-Reise durch die Welt der KI

PROLOG, die vieldiskutierte Computersprache der fünften Generation ist für ST-Benutzer nun auch in der Public Domain Software der ST-Redaktion erhältlich. Was macht diese Sprache so mächtig, daß sie als Programmiersprache im Projekt der fünften Computergeneration eingesetzt werden soll? Lohnt es sich schon wieder mit einer neuen Sprache zu beschäftigen, oder soll man bei seinem geliebten C, Pascal oder Basic bleiben? Und ist das Public Domain Prolog (TOY Prolog) nur ein Spielzeug, wie sein Name anzudeuten scheint? Antworten finden Sie im folgenden Artikel.

Etwas Historie

Die Computersprache der ersten Generation ist die nackte Maschinensprache. In dieser Sprache wurden die ersten Computer direkt durch Setzen von Kippschaltern programmiert. Die Entwicklung symbolischer Assembler hat dann in der zweiten Generation zu einem erheblichen Anstieg in der Produktivität der Programmierung geführt. Für Anwender außerhalb der Informatik wurden Computer erst mit der Entwicklung der Hochsprachen interessant. Eine der ersten Sprachen dieser dritten Generation ist das legendäre, in den Naturwissenschaften immer noch beliebte FORTRAN (**FOR**mular **TRAN**slator). Das aus dem Microcomputerbereich nicht wegzudenkende BASIC (**B**eginners **A**ll purpose **S**ymbolic **I**nstruction **C**ode) ist im übrigen ein vereinfachter FORTRAN Ableger. Mit dem Auftauchen der rekursionsfähigen Sprachen PASCAL (nach Blaise Pascal, frz. Mathematiker) und C (die dritte Version der von D. Ritchie entwickelten Sprache), die die Methode der strukturierten Programmierung unterstützen, war die vierte Softwaregeneration geboren. Allerdings ist der Anstieg in der Produktivität der Programmierung durch die Einführung der Sprachen der vierten Generation längst nicht so dramatisch gewesen, wie noch bei der Einführung der Sprachen der zweiten und dritten Generation. Die in der letzten Ausgabe diskutierte Sprache LISP nimmt hier eine gewisse Sonderstellung ein. Sie gehört neben FOTRAN zu den ältesten Computersprachen und hat dennoch so fortschrittliche Sprachelemente wie Rekursion und funktionale Programmierung zu bieten. Um Mißverständnissen vorzubeugen, möchte ich darauf hinweisen, daß hier nur von der Produktivität des Programmierens die Rede ist, nicht von der Effizienz des laufenden Programms. In dieser Hinsicht dürften auch weiterhin Assemblerprogramme (bzw. Maschinenprogramme) nicht zu schlagen sein. Niemand wird denn wohl auch bestreiten können, daß die Entwicklung eines Algorithmus in einer Hochsprache schneller gelingen wird, als die (genauso mögliche) entsprechende Entwicklung in Assembler. Ein Vergleich in der Programmiereffizienz von BASIC und PASCAL wird da wohl (je nach persönlicher Einstellung) weniger eindeutig ausfallen. C bringt hier gegenüber PASCAL fast keine Vorteile und MODULA, ADA etc. bringen auch keine elementar neuen Sprachkonzepte ein, sondern setzen eher auf Standardisierung von Algorithmen. Ganz anders dagegen der Ansatz von PROLOG, der bewußt vom vielgeliebten Algorithmusbegriff abweicht.

PROLOG – PROgrammieren in LOGik

Mit dieser neuen Programmiersprache der fünften Generation, kommt man dem Traum des Computerbenutzers nahe, ohne Erlernung einer Sprache den Computer nutzen zu können. Dies wird möglich, weil Prolog bereits einen Suchalgorithmus eingebaut hat, der selbständig abläuft, und der dem Benutzer ein gegenüber den Sprachen der vierten Generation völlig neues Lösungskonzept abverlangt. In der Tat sollte man als langjähriger Benutzer algorithmischer Programmiersprachen am besten zunächst alles vergessen was man bisher gelernt hat. Das erworbene Wissen um Algorithmen und Datenstrukturen, den beiden Zauberworten der vierten Generation, behindert den PROLOG Anfänger nämlich eher als das es nutzt. Der Prolog Programmierer braucht sich um den imperativen Ballast seines Problems (wie soll der Computer mein Problem lösen?) also

nicht mehr zu kümmern. Statt dessen deklariert (erklärt) er dem Computer die logische Struktur des Problems (PROLOG ist eine deklarative Sprache), lehnt sich zurück und harrt der Lösungen, die der Computer findet. Deshalb werden wir die Begriffe Algorithmus, Datenstruktur, Prozedur und Funktion zunächst aus dem Gedächtnis streichen und durch neue, dem Programmieren in Prolog besser entsprechende Begriffe ersetzen.

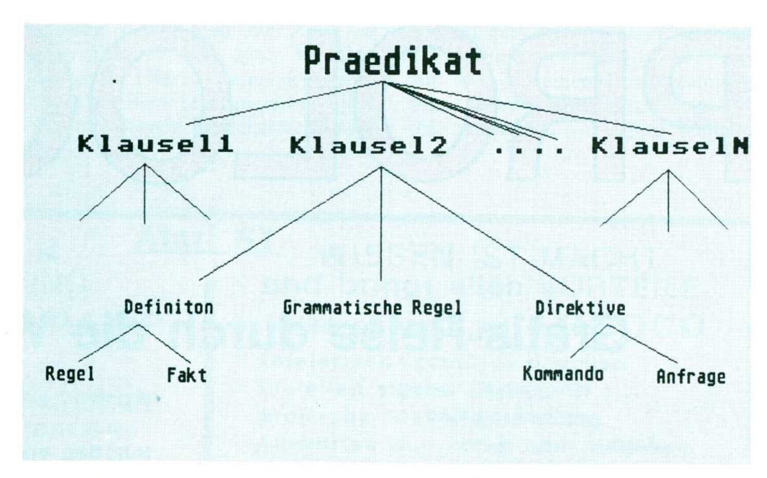


Abb.1: Zusammensetzung von Praedikaten aus Klauseln, Fakten und Regeln

Prädikate, Klauseln, Regeln und Fakten

Grundlage des Programmierens in Prolog sind die Hornschen Klauseln, eine Sonderform der Prädikatenlogik, bei der der Kopf der Klausel (der linke Teil einer Klausel, siehe weiter unten) nur aus dem Namen (dem Funktor) und den Argumenten bestehen darf. Den Aufbau eines Prädikates zeigt Abb. 1. Den gleichen Sachverhalt können Sie auch in der Dokumentation von TOY-Prolog nachlesen, aber dort ist er in der dem Normalverbraucher nicht ganz vertrauten Backus-Naur-Form (BNF) angegeben. Die einzelnen Definitionen, grammatischen Regeln und Direktiven werden mit Hilfe logischer Verknüpfungen (den Junktoren nicht, und, oder; in PROLOG: **not , ;**) zu Klausen zusammengefaßt. Verschiedene Klauseln des gleichen Prädikats werden so interpretiert, daß bei nicht zutreffender Klausel die nächste untersucht wird. Das bedeutet, daß die verschiedenen Klauseln durch oder verknüpft sind. Wenn es nun in der Menge der Klauseln eines Prädikates (alle mit gleichem Namen und gleicher Anzahl von Argumenten) ein einziges gibt, für das die Klausel zutrifft, dann hat also das ganze Prädikat Erfolg. Wie in der Logik auch, können Prädikate, Klauseln etc. nur den Zustand zutreffend (true, Erfolg) oder nicht zutreffend (fail, Mißerfolg) einnehmen. Der Prolog Interpreter arbeitet bei der Zustandsbestimmung eines Prädikates nun nach folgenden Regeln, die man **Resolution** nennt.

1) PROLOG versucht, die verschiedenen Klauseln eines aufgerufenen Prädikates in den Zustand true zu versetzen (Man spricht auch von **Erfolg**). Hierzu beginnt der Interpreter mit der im Programmtext zuoberst stehenden Klausel und arbeitet dann die nächste Klausel des gleichen Prädikates ab, solange es ihm nicht gelungen ist, die gerade bearbeitete Klausel in den Wahrheitszustand true zu versetzen. Hatte der Prolog Interpreter irgendwann Erfolg, wird das Prädikat in den Zustand wahr gesetzt und die evtl. folgenden Klauseln des Prädikates werden nicht mehr bearbeitet (Sie sind ja durch oder miteinander verbunden). **Die Anordnung der Klauseln eines Prädikates ist also unter Umständen sehr bedeutsam.**
2) Hatte Prolog keinen Erfolg mit der Bearbeitung aller Klauseln eines Prädikates, so wird das Prädikat in den Wahrheitszustand fail gesetzt.
3) Zur Bestätigung des Wahrheitsgehaltes einer Klausel übernimmt Prolog selbständig den Rastervergleich (**Unifikation**) und Einsetzung aller Variablen (**Instanzierung**). Existieren mehrere Lösungen, dann werden diese mit Hilfe eines eingebauten Backtracking Mechanismus aufgesucht. Dieser kann vom Benutzer allerdings auch extra unterbrochen werden, etwa wenn man nur an der ersten Lösung interessiert ist.
4) Alle Variablen sind (wie in der Logik üblich) lokal an die Klausel gebunden, in der sie vorkommen.

Das hört sich komplizierter an als es ist. Für Programmieranfänger ist es auch im allgemeinen einfacher mit PROLOG anzufangen, als für Leute, die schon durch den Umgang mit algorithmischen Sprachen „verdorben“ wurden. Man braucht bei den Prädikaten nämlich nur an einen juristischen Vertrag zu denken, der erfüllt ist, wenn eine Klausel des Vertrages zutrifft. In diesem Sinne ist ein Prolog-Programm ein Gebilde aus Verträgen (Prädikaten), die aus einzelnen Klauseln bestehen. Und wie in jedem Vertrag, gibt es auch in PROLOG Regeln, die einen Sachverhalt definieren. Die simpelste Regel ist das Fakt. Es ist eine nicht weiter zu belegende Tatsache.

Der syntaktische Aufbau von Fakten und Regeln

Wie gesagt: das einfachste Prädikat ist das Fakt. Es hat einen Namen (Funktor), der mit einem **Kleinbuchstaben** beginnen muß. Optional ist eine Parameterliste. Jede Klausel muß mit einem Punkt abgeschlossen werden, also auch das Fakt.

Beispiele für Fakten:

atari. vater(august). vater(august,emil). kind(emil).

Die Namen der Fakten (atari, vater, august, emil, kind) sind nicht weiter strukturiert und heißen deshalb wie in LISP Atome. Wie man sieht, kann der gleiche Funktor „vater“ für zwei verschiedene Fakten benutzt werden. So läßt sich vater(august) als das Fakt interpretieren, daß ein gewisser Herr mit dem Namen august Vater eines Kindes ist. Das Fakt vater(august,emil) hingegen hält fest, das ein gewisser Herr august Vater von emil ist. Ob-

wohl beide Aussagen verwandt scheinen, so sind sie für Prolog doch gänzlich verschieden. Wir müssen bei der Benutzung eines Prädikates stets die Anzahl der Parameter (die Stelligkeit oder Arität) des Prädikates berücksichtigen. Also sind vater/1 (/1 ist die Stelligkeit des Prädikates. Beispiel: vater(august).) und vater/2 (Stelligkeit 2. Beispiel: vater(august,emil).) zwei völlig verschiedene Prädikate, trotz gleichen Namens (Funktor). Die Stelligkeit von atari in der obigen kleinen Datenbank ist demnach 0. Natürlich enthält die Vater-Kind Beziehung in der kleinen Beispieldatenbank überflüssige Informationen. Beispielsweise ist bereits aus dem Fakt vater(august, emil) klar, daß emil das Kind von august sein muß. Also wird man die Datenbank mit den Fakten zu einer Wissensbank mit Fakten und Regeln machen, indem man das Prädikat kind aus dem Prädikat vater/2 mit Hilfe einer Regel schließt:

Regel:
X ist ein Kind, wenn das zweite Argument des zweistelligen Prädikates vater mit X in Übereinstimmung gebracht (instanziert) werden kann.

Oder anders ausgedrückt, wenn es eine Person Y gibt, die mit X zusammen im Fakt vater/2 in der Datenbank steht (vater(august,emil).), dann ist X das Kind von Y. Hier übernehmen X und Y die Funktion von Variablen, d. h. sie stehen stellvertretend für ein Objekt. Bitte beachten Sie: **in PROLOG beginnen alle Variablennamen mit einem Großbuchstaben.** Im Gegensatz zu den imperativen Sprachen der vierten Generation lassen sich Variablen keine Werte **zuordnen**. Diese werden in PROLOG vielmehr **instanziert** (vom Interpreter im Rahmen eines Muster- oder Rastervergleiches zugewiesen), d. h. innerhalb einer Klausel kann eine Variable nur einmal einen Wert erhalten, danach behält sie diesen bis zum Verlassen der Klausel (dem Gültigkeitsbereich der Variablen). Diese **referentielle Transparenz** ist es auch, die im übrigen PROLOG für die Computerbauer der 5. Computergeneration so interessant macht, weil sie **parallele Datenverarbeitung** ermöglicht. Die Regel für das Kind ließe sich in PROLOG folgendermaßen formulieren:

```
kind(X) :- vater(Y,X).
```

Hierin bedeutet kind(X) den **Regelkopf** und (durch :- getrennt) vater(Y,X) den **Regelrumpf**. Man liest die Regel am besten so: Es gilt: X ist kind, wenn vater von Y und X gilt. Fragen wir nun nach dem Prädikat kind(emil), so wird uns PROLOG **yes** antworten, weil der Interpreter in der Datenbank das Prädikat vater/2 findet. Er bringt nämlich die Variable Y mit dem Fakt august in Übereinstimmung und den Inhalt der Variablen X aus dem Kopf (emil) mit dem Inhalt der Variablen X aus dem Fakt der Datenbank (emil). Da der Rastervergleich Erfolg hat, hat auch das Prädikat vater/2 Erfolg. Und somit auch das Prädikat kind(emil). Die im Verlauf des Rastervergleiches in Übereinstimmung gebrachte Varialbe Y (august) unterschlägt Prolog, weil diese Variable nicht im Prozedurkopf, also in der Anfrage (dem **goal**) enthalten war.

Unifizierung und Instanzierung

Diese beiden Begriffe wurden bisher schon erwähnt und sollen nun etwas näher erläutert werden. Es sind die Fachausdrücke für einen Muster- oder Rastervergleich von Datenstrukturen in Prolog und die damit verbundene Zuordnung von Daten an Variable. Wie bereits gesagt unterscheidet sich Prolog von den imperativen Sprachen der vierten Generation u. a. durch den Gebrauch von Variablen, die im Sinne der Logik Platzhalter für irgendetwas darstellen. Demnach gibt es in Prolog auch keine Datenstrukturen, die (wie in Pascal oder C) deklariert werden müssen und über deren korrekte Verwendung ein strenger Typencheck wacht (Es sei an dieser Stelle ein bissiger Seitenhieb auf Turbo Prolog aus dem MS-DOS Lager erlaubt, welches durch die Hintertür diesen Vorteil wieder zunichte macht). Vielmehr kann jede Prolog Variable mit jedem legalen Prolog Term instanziert werden. Die **Unifizierung** verwendet die **Instanzierung** von Variablen lediglich als Elementaroperationen. Unter **Unifizierung** versteht man also einen allgemeinen Mustervergleich zweier Prolog Terme. Hierbei gibt es keinen „L-Wert" und „R-Wert", d. h. die beiden Prolog Terme die unifiziert werden sollen sind gleichberechtigt. Eine Unifizierung kann dann drei Ergebnisse haben:

1) Beide Terme sind bereits gleich, d. h. sie haben sowohl die gleiche Struktur, als auch die gleichen Konstanten, Unterstrukturen oder Variablen Einsetzungen an den gleichen Stellen. Dann beschränkt sich der Rastervergleich auf diesen Vergleich und ist damit erfolgreich.

2) Es gibt Unterschiede zwischen den Termen. Diese können aber dadurch beseitigt werden, das in einem oder beiden Termen geeignete Variablen mit Fakten oder Termen aus der Wissensbank in Übereinstimmung gebracht werden. Prolog führt dann diese Einsetzungen durch und der Rastervergleich schließt erfolgreich ab.

3) Können die Unterschiede nicht wie in 2) beseitigt werden, dann werden keine Einsetzungen vorgenommen und der Rastervergleich endet erfolglos.

Abb. 2 zeigt einige der möglichen Fälle.

Listen in PROLOG

Wie in LISP, so sind auch in PROLOG die Listen ein ganz wesentliches Datenelement. Im Gegensatz zu Lisp, werden in Prolog die Listen in eckige Klammern gesetzt. Die leere Liste schreibt man natürlich als [ ]. Die einzelnen Listenelemente werden durch Komma getrennt. Folglich ist **[bar, casino]** eine Liste mit atomaren Konstanten, die die zu vermeidenden Orte im Labyrinth von Listing 3 enthält. Eine nicht leere Liste hat stets die Form **[kopf | rest]**. Hierin bedeutet **kopf** das erste Element der Liste (in LISP: **CAR**) und **rest** bedeutet eine Liste (notfalls die leere Liste) mit den restlichen Elementen der Liste (in LISP: **CDR**). Im eben genannten Beispiel wäre **rest** also gleich **[casino]**. In Prolog Notation würde man sagen, **kopf** ist mit **bar** instanziert und **rest** mit **[casino]**. Auf die leere Liste paßt natürlich nut [], und der Versuch **[kopf|rest]** mit der leeren Liste zu instanzieren würde mißlingen.

Programmieren in PROLOG

Bevor ich mit der Besprechung des PD Prolog beginne, noch einige Anmerkungen zum generellen Umgang mit Prologprogrammtexten. Jedes Programm besteht aus einer Sammlung von Prädikaten, die sich gegenseitig aufrufen. Im allgemeinen sind in einem Prolog Programm eine große Anzahl von Fakten als Wissensbasis vorhanden. Die anderen Prädikate sind meistens Regeln (Für die Erläuterung von grammatischen Regeln und Direktiven reicht hier der Platz nicht aus). Eines dieser Prädikate übernimmt den Start des Programms und sorgt für den eigentlichen Ablauf. Nehmen wir als Beispiel das einstellige Prädikate vater/1. Nehmen wir ferner an, daß folgende Datenbank vorliegt:

vater(august). vater(otto). vater(eduard).

Also sind august, otto und eduard Väter. Beachten Sie die Kleinschreibung, die august, otto und eduard als atomare Konstanten ausweist. Stellen wir nun die Frage, wer alles als Vater bekannt ist, so müssen wir ja alle Ergebnisse von august bis eduard erhalten. Dies besorgt der in Prolog ständig aktive Backtrackingmechanismus. Das Protokoll der Anfrage sähe dann etwa so aus (eigene Eingaben **fett**, Antworten des Computers *kursiv*.):

vater(X).

X= august ;	% Das Semikolon ist
X= otto ;	% vom Benutzer ein-
X= eduard ;	% gegeben. Als logi-
no	% sches „ODER“ for-
?-	% dert es den PRO-
	% LOG Interpreter
	% auf Alternativen zu
	% finden. Das no be-
	% deutet: keine weite-
	% ren Alternativen.

Das bedeutet, daß der Prolog Interpreter nach Erhalt des Semikolons in die Suchprozedur für das Prädikat vater/1 erneut zurückgeschickt wird, um weitere Einsetzungen für die Variable X zu finden, d. h., daß alle in dem entsprechenden Prädikat vorgenommenen Einsetzungen ungültig werden. Abb. 3 zeigt die aus [8] entnommene Skizze des Boxenmodells eines Prolog Prädikates. Das Prädikat wird aufgerufen und hat die Möglichkeit des Erfolges oder Mißerfolges. Im ersten Fall verläßt der Prolog Interpreter das Prädikate durch den **Erfolg** Ausgang und ruft ggf. das nächste Prädikat auf. Im zweiten Fall wird der **Mißerfolg** Ausgang benutzt. Dieser führt (falls vorhanden) zu dem Prädikat zurück, von dem das momentane Prädikat gerufen wurde. Das vorhergehende Prädikat wird dabei durch den **Nochmal** Ein-

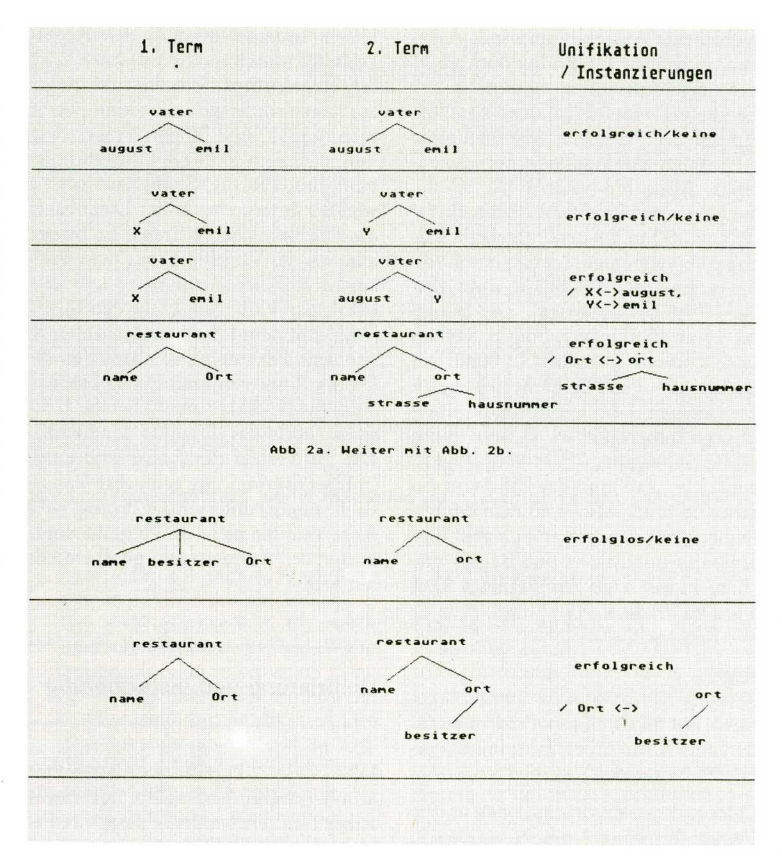


Abb. 2: Beispiele von Unifizierungen

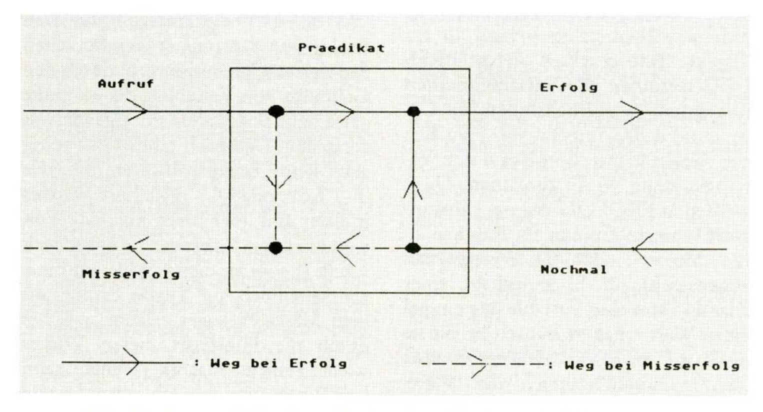


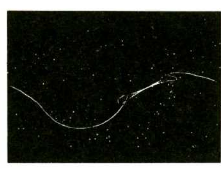
Abb. 3: Boxenmodell eines Prolog Praedikats nach Schnupp.

DAS SONNENSYSTEM –

spektakulär wie nie zuvor:

SKYPLOT

DAS ASTRONOMISCHE SUPERPROGRAMM FÜR DEN MEGA-ATARI


Die Bahn des Kometen Halley

Informationen über die Planeten unseres Sonnensystems, so z. B. über die Oortsche Wolke, die jenseits des Planeten Pluto im Weltall entdeckt worden ist. Aus diesen fernen Regionen stammen jene Kometen wie der Halley'sche Komet, der derzeit im Anflug in Richtung Erde ist.

Karte umfasst 39.65 x 23.48 Bogensekunden
Mond
Entfernung von der Erde 398250.1 km
Ekliptikale Laenge 305.38, Breite -1.13 Grad
Rektaszension 20.33 h, Deklination -20.54 Grad
Phasenwinkel 151.15 Grad
Mondalter 2.4 Tage
Scheinbarer Aequatordurchmesser 1800.61 Bogensekunden

Dieses Programm verrät Ihnen alles, was Sie wissen wollen. Nicht nur die Sonne und deren Finsternis, die Milchstraße oder der Saturn werden Sie faszinieren, nein, auch die „Schönheit" des Himmels: der Hantel-Nebel aus dem Sternbild Vulpecula; insgesamt 612 Sterne, 1053 Nebel und Sternhaufen und alle Planeten. Skyplot zeigt Ihnen Ihr Sternzeichen am Himmel, wie es sich zeigt, welche Sterne und Planeten diesem angehören und wo es sich befindet.



Die Menüleiste gibt Ihnen einen Überblick über die Vielfalt von Skyplot:





Suchen nach Sternbildern

Möchten Sie wissen, wann die nächste Mondfinsternis ist?

Möchten Sie wissen, wann der Halley'sche Komet sich der Erde nähert?

Möchten Sie wissen, wie sich der Himmel an den verschiedenen Jahreszeiten zeigt?

Möchten Sie mehr wissen über das Ungewisse?

Bestellen Sie jetzt, und Sie können an einer faszinierenden Fülle von Informationen über unser Sonnensystem teilhaben und die Antwort auf viele Fragen erfahren.

BESTELLCOUPON

Einsenden an: **CSM Computer Special Marketing GmbH**
Vorhölzerstraße 18, 8918 Dießen

Hiermit bestelle ich das Programm **Skyplot** auf Diskette zum Micro-Preis von **149,– DM**

Name:
Vorname:
PLZ/Ort:
Straße:

Versandwunsch bitte angeben:
○ per NN
○ per Vorkasse
○ per Verrechnungsscheck

gang betreten, da dieses Prädikat ja offensichtlich schon einmal Erfolg hatte (sonst wären wir nicht im darauffolgenden Prädikat gelandet). Der **Nochmal** Eingang wird nicht nur bei Mißerfolg des nachfolgenden Prädikates betreten, sondern auch dann wenn (wie im obigen vater/1 Beispiel) durch Eingabe eines „;" vom Benutzer die Suche nach alternativen Lösungen verlangt wird. Deshalb ist der Weg in den **Nochmal** Eingang in Abb. 3 auch nicht gestrichelt gezeichnet. Abb. 4 zeigt die Lösungssuche des Prolog Interpreters bei der Suche nach Lösungen zu **vater(X)** im Boxenmodell. Aus Platzgründen verzichte ich hier auf die Erläuterung von Produktionsregeln. Diese sind für den Anfänger nicht ganz einfach zu verstehen. Eine sehr gute Einführung findet sich im Buch von Schnupp [8].

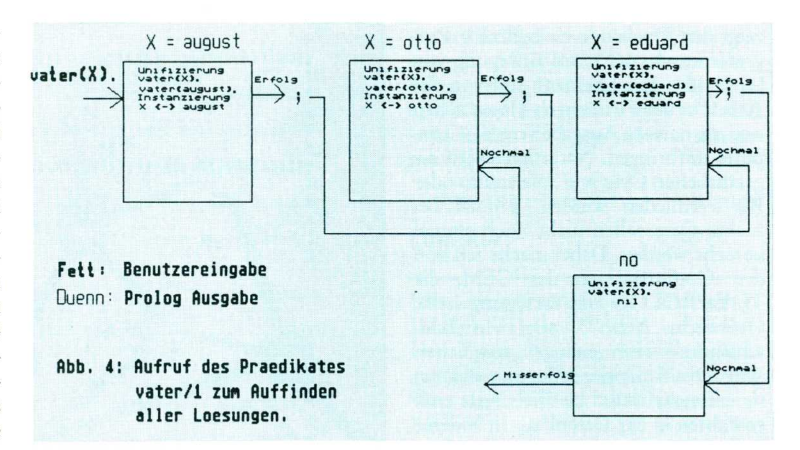


Abb. 4: Aufruf des Praedikates vater/1 zum Auffinden aller Loesungen.

Umgang mit dem PD Prolog

Die ersten Gehversuche sollte man sicherlich direkt im Prolog Interpreter versuchen. Also klickt man **TOY.PRG** an. Damit startet der innere Interpreter, der nach Angaben des Autors in C und Assembler geschrieben ist. Die Originalversion des TOY-Prolog ist übrigens in Pascal geschrieben und ist bei den Verfassern [9] nachlesbar. Anschließend wird der äußere Interpreter geladen, der im File **SYSFILE.TOY** vorliegt. Man erkennt das Laden daran, daß jetzt der Schriftzug **Booting...** und das Logo auf dem Bildschirm erscheinen. Anschließend meldet sich Prolog mit der Meldung **TOY – Prolog listening ?-**. Das **?-** signalisiert Prologs warten auf Befehle. Versucht man nun das einfache vater/1 Prädikat nachzuvollziehen, so muß man zunächst die Datenbasis eingeben. Dies ermöglicht das Prädikat **assert**/1 oder **assertz**/1. Beide nehmen das Argument (ein legaler Prolog Term) in die Datenbank auf. **assert**/1 fügt das Argument an den Anfang der Datenbank hinzu, **assertz**/1 dagegen an das Ende. Im Anhang der Dokumentation ist übrigens die Tastenbelegung beschrieben, die im **SYSFILE.TOY** bereits vorgenommen ist. Und so entfällt auch das lästige Tippen von assert (vater(august)). etc. Man drückt die **Insert**-Taste und schon steht **assert(** auf dem Bildschirm, bereit zur Aufnahme eines Prolog Termes in die Datenbank (an den Anfang). Die Tastenbelegung finde ich ganz praktisch, aber wer die Tastenbelegung verändern möchte, hat die Möglichkeit das im **SYSFILE.TOY** mit Hilfe eines Editors zu tun. Nach Eingabe der Datenbank (abschließenden Punkt nicht vergessen!) kann man dann die verschiedenen Lösungen von **vater(X).** durch Eingabe des „;" anfordern. Man wird schnell feststellen, daß der Umgang mit dem Interpreter nicht gerade bequem ist. Zur Erstellung einer umfangreichen Wissensbasis wird man deshalb schnell zu einem Editor greifen, in dem der Text editiert werden kann. Das Einlesen der Wissensbasis in den Interpreter erfolgt dann mit Hilfe des **consult**(Filename). Prädikates. Das Ende des Lesevorganges erkennt Prolog allerdings nicht am EOF-Code des Files. Man muß dem Prädikat consult/1 das Ende des Lesevorganges ausdrücklich durch **end.** bekanntgeben. Dann erst erscheint wieder das Prolog Prompt **?-** und Prolog ist zur Beantwortung unserer Anfragen bereit. Die Syntax des PD-Prolog ist nahezu identisch mit der Syntax des DEC-10 Dialektes, dem de facto Standard. Die geringen Abweichungen sind in der Dokumentation festgehalten und betreffen nur wenige Prädikate (z. B. **pname**). Trotzdem wird der Anfänger beim Nacharbeiten der Literatur oder des nachfolgenden Beispiels feststellen, daß eine Benutzer-Shell angenehm wäre. Natürlich sollte man den Editor, den Prolog Interpreter und die ***.TOY** Files in eine RAM-Disk kopieren, wobei die PD-Diskette Nr. 5 mit ihrer RAM-Disk und ihrem COPY Programm hervorragende Dienste leistet.

Der Entwicklungszyklus sieht dann folgendermaßen aus:

1) Aufruf von **ED.TOS** (o. ä.). Eingabe des Sourcefilenamens. Eingabe des Programmtextes.
2) Abspeichern des Programmtextes.
3) Anklicken des **TOY.PRG** Files.
4) Konsultieren des Sourcefiles.
5) Beenden des Prolog Dialoges mit **stop.**

Während der Entwanzungsphase wäre es angenehm, sich das lästige Eintippen des Sourcefilenamens und das Konsultieren desselben zu ersparen. Die im LISP Artikel [7] beschriebene Benutzershell **MENU+** erfüllt den ersten Teil des Wunsches, da aber TOY ein PRG File ist, akzeptiert es keine Argumentzeile, wie die TTP Programme. Ich habe deshalb den äußeren Interpreter in **SYSFILE.TOY** so geändert, daß beim Starten von TOY. PRG die im File **DEFAULT** stehende Datei automatisch beim Start von TOY.PRG konsultiert wird. Listing 1 und Listing 2 zeigen die entsprechenden Änderungen im **MENU.INF** und im **SYSFILE.TOY** File. Die Hinweise im Listing 1 sind zwar nur für die Besitzer des **MENU+** Programms interessant, Listing 2 gibt aber vielleicht den Benutzern anderer Shells ein paar nützliche Hinweise.

Lernen am Beispiel

Ich habe sicherlich von den Realtime-Programmierer eine Menge Geduld verlangt, aber jetzt geht's log. Listing 3

zeigt den Quellcode eines Prolog Programmes, das auf einer Anregung von [4] beruht. Ein Ehemann soll von der Arbeit in der Firma nach Hause gehen und die neueste Ausgabe von ST-Computer mitbringen. Natürlich sollen die gefährlichen Orte wie Spielcasino oder Bar vermieden werden. Einmal besuchte Orte sollen nicht noch einmal besucht werden. Dabei mache ich von den Grafikroutinen des GEM, die TOY-PROLOG zur Verfügung stellt, Gebrauch. Abb. 5 zeigt ein Bildschirmfoto vom einzigen möglichen Weg. Der Anfänger sollte hier Phantasie entwickeln und beispielsweise einige Fakten in der Datenbasis in anderer Reihenfolge eingeben. Um das Straßennetz zu ändern, brauchen lediglich andere Koordinaten für die Orte eingegeben zu werden. Natürlich läßt sich auch die topologische Struktur des Graphen ändern, indem man die Fakten **straße/1** ändert. Allerdings sollte man die Koordinaten von **ort (firma,__,__)** und **ort(wohnung,__,__)** nicht ändern, da sonst die Anordnung der Start- und Zielpfeile mit verändert werden müßte. Für eine ausführliche Besprechung des Programms ist hier leider kein Platz.

Natürlich benötigt man in Prolog keine Zeilennummern. Daß die Listings trotzdem damit versehen wurden dient lediglich zur besseren Orientierung beim Abtippen und soll helfen Fehler zu vermeiden.

```
###################################
#                                 #
# Standard C MENU+ info file      #
#                                 #
# Copyright (c) 1986 Metacomco plc #
#                                 #
###################################
#
# Tools menu item definitions
#
# Form: <item name> = <command line definition>
#

TOOLS

EDIT          =  {command_dir}\ED.TTP {path}\{file}.{type} {editor_opts}
PROLOG        =  {command_dir}\TOY.PRG

#
# File menu item definitions
#
# Form: <item name> = <file pattern>
#

FILE

Choose      =  *.*
Choose PRO  =  *.PRO

#
# Option menu item definitions
#
# Form: <option name> = <initial value>
#

OPTIONS

CURRENT_DIR    = C:
PATH           = C:
COMMAND_DIR    = C:
EDITOR_OPTS    =
COMPILER_OPTS  =
```

```
'ear' :
   'see'('start').
   'nl' .
   'display'('TOY Prolog listening:') .
   'nl' .
   'tag'('loop') . []
```

Listing 2a: Änderungen im File SYSFILE.TOY.

```
see('c:default'),read(Datei),consult(Datei),seen.
```

Listing 2b: Inhalt des Files START, der von TOY-Polog während es bootens gelesen wird.

Eigenschaften des PD Prolog

Das TOY Prolog ist eine recht vollständige Implementation des DEC-10 Dialektes. Der größte Teil der Implementation ist dabei im äußeren Interpreter untergebracht, so daß die Geschwindigkeit nicht berauschend ist. Die rekursive Quicksort Routine (Listing 4) benötigte für das Sortieren einer Integer Liste mit 100 Elementen 12 s und liegt damit im Bereich des XLISP Interpreters (siehe [7]) aus der PD. Viel störender ist dagegen die Beschränkung auf Integer Arithmetik. Natürlich ist PROLOG (wie LISP) keine Sprache die vornehmlich für arithmetische Datenverarbeitung gedacht ist. Das schreiben auch Clocksin und Mellish [1] in ihrem Standardwerk. Dennoch ist eine Gleitkommaarithmetik aus einer Sprache nicht wegzudenken. So soll die angekündigte neue Version von MPROLOG für den ST u. a. auch einfach genaue Gleitkommaarithmetik (letzte Information: auch doppelt genaue Darstellung) erlauben. Das System besitzt einen Debug-modus und einzelne Prädikate lassen sich mit dem Prädikat **spy/1** auch in ihrer Wirkung ganz gut beobachten, da aber TOY-Prolog alle Variablen in interner Schreibweise ausgibt, habe ich es als übersichtlicher empfunden auf die Benutzung des Debuggers zu verzichten und bei Bedarf im Programm eigene Fehlermeldungen unterzubringen. Das ist besonders Anfängern zu empfehlen, die den Weg eines Prolog Programmes nachvollziehen wollen. Ansonsten aber hat man ein komplettes Prolog System vor sich, mit dem sich unter Benutzung eines eigenen Editors und einer eigenen Shell und der RAM-Disk und dem COPY Programm aus der PD recht komfortabel arbeiten läßt, wenn man

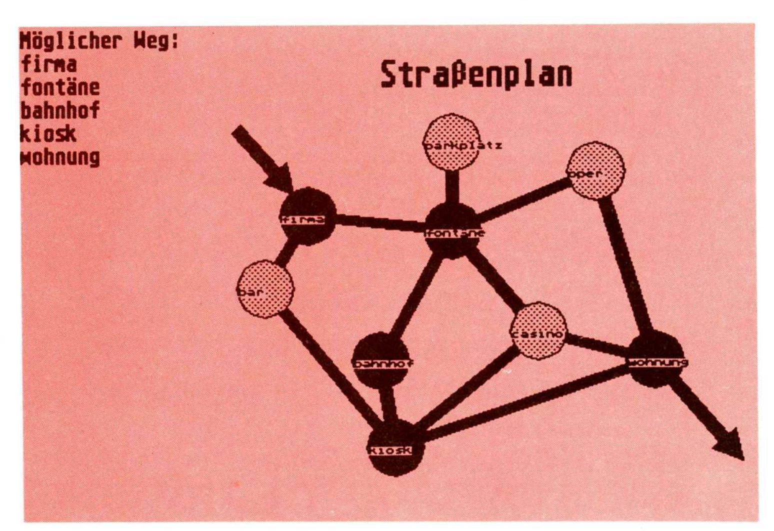


Abb.5: Problem des heimkehrenden Ehemannes(-frau?)

```
%Problem des heimkehrenden Ehegatten

%Fakten über die Orte

ort(wohnung, 440, 230).
ort(oper, 400,100).
ort(casino, 360, 210).
ort(parkplatz, 300, 80).
ort(fontäne, 300, 140).
ort(bahnhof, 250, 230).
ort(bar, 170, 180).
ort(kiosk, 260, 290).
ort(firma, 200, 130).

%-----------------------------------------------------------------------

%Fakten über die Straßen.

straße(firma,bar).            straße(bahnhof,kiosk).
straße(firma,fontäne).          straße(casino,kiosk).
straße(bar,kiosk).             straße(casino,wohnung).
straße(fontäne,bahnhof).             straße(oper,wohnung).
straße(fontäne,casino).         straße(kiosk,wohnung).
straße(fontäne,oper).
straße(fontäne,parkplatz).

%-----------------------------------------------------------------------

vermeide([bar,casino]).     %Ist besser!

%-----------------------------------------------------------------------

gehe(Hier,Da) :-
   vermeide(Gefahren),
   pfad(Hier,Da,Gefahren,[Hier]).

%-----------------------------------------------------------------------
pfad(Hier,Da,Gefahren,Spur) :-
   Hier == Da,
   member(kiosk,Spur),
   bell,
   write('Möglicher Weg:'), nl,
   reverse_write(Spur),
   rch, !,
   fail.
pfad(Hier,Da,Gefahren,Spur) :-
   (
      straße(Hier,Zwischen)
   ;
      straße(Zwischen,Hier)
   ),
```

die in Listing 2 angegebenen Änderungen am **SYSFILE.TOY** durchführt. Sonst ist die Bedienung umständlich, aber dem Erlernen von PROLOG steht nichts im Wege außer einem guten Lehrbuch.

Literatur

Im Literaturverzeichnis sind einige Lehrbücher über Prolog aufgeführt. Dem Prolog Neuling, der zudem Wert auf eine praxisbezogene Einführung Wert legt, sei das 1986 erschienene Buch von **Schnupp** [8] empfohlen. Der Autor erläutert an Hand praxisorientierter Beispiele die Vorteile von Prolog. Theoretische Erörterungen bleiben auf ein absolutes Minimum reduziert. Dies ist wohl das am leichtesten verständliche deutschsprachige Buch über Prolog. Sehr positiv ist der übersichtliche Anhang zu werten, in dem die Syntax der DEC-10 Prolog Prädikate definiert und an Beispielen erläutert wird. Der Prolog Neuling hat hier eine übersichtliche Informationsquelle, die erheblich mehr Informationswert besitzt als die sehr axiomatische BNF aus der Dokumentation in der PD.

Wer sich in Prolog schon etwas eingearbeitet hat, dem sei das Buch von Giannesini [5] empfohlen. Man sollte sich allerdings vom Eingangsthema des Buches (der Speisekarte eines Restaurants) nicht täuschen lassen. Danach kommt es ziemlich dick theoretisch. Und das auch noch in der Syntax von PROLOG II, dem Dialekt, welcher der Nachfolger des Ur-Prolog der GIA (Groupe d'Intelligence Artificielle) Unitersität von Aix-Marseille [3] ist. Dem an der Logik interessierten Programmierer hat dieses Buch natürlich mehr zu bieten, zumal es die wesentlichen Begriffe aus der Logik (sofern sie benötigt werden) zu Beginn des jeweiligen Kapitel definiert. Beide deutschsprachigen Bücher enthalten im Anhang noch einen Hinweis auf die jeweils anderen Prolog Dialekte DEC-10 bzw. Prolog II und micro-Prolog.

Schon aus historischen Gründen sollte man auf die Lektüre des Klassikers von Clocksin und Mellish nicht verzichten [1]. Dieses Buch hat immerhin auch den in TOY-Prolog verwendeten DEC-10 Dialekt definiert.

Schließlich möchte ich noch auf das Buch von Kluzniak [9] hinweisen, auf dem die vorliegende ST Version beruht und welches ein komplettes Source Listing (in Pascal) für den inneren Interpreter enthält. Hier findet man eine sehr ausführliche Anatomie des vorliegenden Prolog Interpreters, die in Bezug auf Sorgfalt und Vollständigkeit keinen Wunsch mehr offenläßt.

Ein ganz besonderer Leckerbissen ist das Buch von Coelho [2]. Hier findet man ein breites Spektrum von Prolog Programmen aus allen möglichen Interessengebieten zusammengetragen. Leider ist es nicht ganz einfach zu erhalten.

Fazit

Wie ich bereits früher erwähnt habe [6], ist Prolog eine der Computersprachen, die für das FCGS-Projekt gewählt wurden. Einer der Gründe ist die eingangs erwähnte **referentielle Transparenz** von Prolog. Diese Eigenschaft ist die Grundvoraussetzung für den Einsatz von paralleler Datenverarbeitung, damit sichergestellt ist, daß sich der Wert einer Variablen im Prozeß 1 nicht ändert, während sich parallel dazu Prozessor 2 im Prozeß 2 mit der gleichen Variablen beschäftigt. Und da die Entscheidungsbäume im Bereich der künstlichen Intelligenz rasch in den Himmel wachsen, ist neben der Einführung von Heuristiken (auf Erfahrung beruhenden Verfahren) die Entwicklung von Multiprozessorsystemen mit Parallelverarbeitung eine absolute Notwendigkeit und im FCGS-Projekt vorgesehen. Es ist daher wohl nicht zu umgehen sich mit Prolog zu beschäftigen, will man nicht den Anschluß an die kommende Entwicklung verlieren. Zudem macht diese Prolog Implementation durch die Einbindung der GEM-Routinen außerordentlich viel Spaß. Bleibt zum Schluß nur die definitive Festlegung des Leserprädikates **clever**/1:

clever(Leser) :– liest(Leser,st),
programmiert__in(Leser,prolog).

Mit anderen Worten: unverzüglich eine Diskette formatieren und zusammen mit DM 5,– plus Spesen an die ST-Redaktion schicken und die PD Diskette Nr. 11 verlangen. Besonders erfreulich ist natürlich auch die Ankündigung zweier kommerzieller Pro-

```
      not member(Zwischen,Gefahren),
      not member(Zwischen,Spur),
      lösche([Zwischen|Spur]),
      zustand([Zwischen|Spur]),
      (   modus(einzelschritt), rch; not modus(einzelschritt) ),
      pfad(Zwischen,Da,Gefahren,[Zwischen|Spur]).

%-----------------------------------------------------------------------
%Markiert die besuchten Orte im derzeitigen Suchzustand

zustand([]).
zustand([Kopf|Rest]) :-
   zustand(Rest),
   ort(Kopf,X,Y),
   grf_f_type(1),
   kreis(X,Y), sum(X1,18,X), grf_text(X1,Y,Kopf).

%-----------------------------------------------------------------------
%Löscht die nicht besuchten Orte

lösche(Spur) :-
   ort(Name,X,Y),
   not member(Name,Spur),
   grf_f_type(2),
   kreis(X,Y), sum(X1,18,X), grf_text(X1,Y,Name),
   fail.
lösche(_).

%-----------------------------------------------------------------------
%Ausgabe einer Liste in umgekehrter Richtung

reverse_write([]).
reverse_write([Kopf|Rest]) :-
   reverse_write(Rest),
   write(Kopf), nl.

%-----------------------------------------------------------------------
%Aufruf dieses Prädikates startet das Programm!

start(Modus) :- assert(modus(Modus)), start.

start :- cls, grf_mode, grf_f_type(2), grf_f_style(1), logo,
         grf_t_height(4), verbindung, karte, gehe(firma,wohnung).

%-----------------------------------------------------------------------

logo :- grf_t_height(20), grf_text(250,20,'Straßenplan').

%-----------------------------------------------------------------------
%Dieses Prädikat zeichnet einen Kreis mit dem Radius 20

kreis(X, Y) :- grf_circle(X, Y, 20).

%-----------------------------------------------------------------------

%Dieses Prädikat zeichnet den Stadtplan

karte :- ort(Name, X, Y) , kreis(X, Y) , sum(X1,18,X),
         grf_text(X1,Y,Name), fail.
karte.

%-----------------------------------------------------------------------

% Zeichnet die Straßenverbindungen

verbindung :-
   grf_l_width(7), grf_l_ends(0,1), grf_pline([150,70,190,115]),
   grf_pline([440,230,500,300]), grf_l_ends(0,0),
   straße(Von,Nach), ort(Von,X1,Y1), ort(Nach,X2,Y2),
         grf_pline([X1,Y1,X2,Y2]), fail.
verbindung.

%-----------------------------------------------------------------------

:- cls, write('Starten des Programms mit "start."'), nl,
       write('Nachdem ein Weg gefunden ist, meldet sich die Glocke.'), nl,
       write('Wollen Sie im Einzelschrittverfahren die Lösungssuche verfolgen,'), nl

       write('dann geben Sie "start(einzelschritt)." ein.'), nl,
       write('Sie müssen dann vor jedem Schritt die "RETURN" Taste drücken.'), nl.


end.
```

Kieckbusch-Window

Depotverwaltung 2.5 .... 498 DM

Für Aktien, Optionsscheine, Optionen. Gewinn & Verlustberechnung. Druckt Depotübersichten. Vier Liniencharts, zwei Langfristcharts. Point- & Figurchart. Frame-Grabbing. Ex-Dividende & Kapital-Erhöhung, auch Operation Blanche. Berechnung persönl. Steuersatz. Programm von Börsen-Insider umfänglich und sinnvoll geschrieben.

Multi-Hardcopy .... 98 DM

*Anpassung an wirklich **jeden** Drucker, ob 8, 16 oder 24 Nadeln, ob Farb- oder Laser-Drucker. Ausdruck in Hoch- & Querformat. Ausschnitt-Druck auch gespiegelt, vergrößert, invers usw. Spooler-Betrieb. Formate: Degas / Neo / Doodle / Art Director. Diverse Drucker-Treiber im Lieferumfang - Selbsterstellung problemlos.*

MAC-Emulator .... 498 DM

Das bewährte Produkt – durch Großeinkauf jetzt preisreduziert.

Typesetter & Megafont von XLent

*149 DM / 119 DM beides in **deutsch**!!!*
Zum Erstellen des Layouts und zum Ausdrucken
Klein-Desktop Publishing!!!

VIP Professional – deutsch – GEM!!!

Vorsicht: es werden billig amerikanische Grauimporte angeboten, die auf deutschen ST nur unvollständig funktionieren.

Wir helfen Ihnen auch dann, wenn Sie solch ein Produkt erworben haben und es nun Probleme gibt. Rufen Sie uns an!

Wir bieten: deutsches Programm, deutsches Handbuch, Hotline, Umtausch- & Update-Service sowie Schulung und VIP-Club.

Desk-Assist II+

Das Rundumprogramm für Ihren Atari ST

Terminplanung, Kalender, Uhr (auch in der Menüzeile)
Alarmtermine (Anzahl unbegrenzt), Notizblatt
Rechner (dez/hex/bin/Zeit/Datum/Maßumrechnung/18-stellig)
residente Adressen und Telefondatei mit komfortablen
Such- und Druckmöglichkeiten
Druckerspooler und Hardcopy (auch Teilbild)
ausgefeilte Drucker-Anpassungs-Möglichkeiten
Verschlüsselung beliebiger Dateien
Super-Terminalprogramm im Lieferumfang!!!
zuverlässiges deutsches Produkt

*Preis: nur **198, – DM***

Sonderangebot:

VIP Professional plus Desk-Assist II+

*zusammen für **848, – DM***

Computer-Technik Kieckbusch GmbH
Am Seeufer 11+22 · 5412 Ransbach
Tel. 0 26 23/16 18 · FAX: 0 26 23/21 40

Art & Film Director

Wir liefern Programm und Handbuch natürlich in DEUTSCH und ab Lager

Art: 209 DM / Film: 235 DM

Damit alle ST-Besitzer diese unübertroffenen Programme nutzen können:

Große Inzahlungsnahme-Aktion

Wir nehmen jedes Original-Malprogramm für

volle 50 DM

in Zahlung, wenn es komplett eingeschickt wird.

Achtung: Art & Film laufen nur in Farbe!!!

Rechenblatt ist da!!!

Deutsches Kalkulationsprogramm mit 300 Spalten á 1000 Zeilen.
Voll unter GEM. Leicht, logisch, sofort bedienbar.
50 Mathematische Funktionen, 70 Befehle über die Maus,
Notizblatt und Taschenrechner aufrufbar.
Übernahme des Taschenrechner-Ergebnisses ins Rechenblatt!
10 Macros auf den Funktionstasten
Datenbank-Sortierfunktionen enthalten.

Unser bewährter Produkt-, Informations- und Hotline-Service findet auch hier wieder Anwendung und ist selbstverständlich im Preis enthalten:

148, – DM

Sonderangebot: „Rechenblatt" plus „Desk Assist" für 350 DM!!!

Sonderangebots-Liste

OSS-PASCAL, das Original, englisch .... 100 DM

***Haba Writer II+**, Textprogramm, deutsch .... 80 DM*

***Haba View**, Datenbank, deutsch .... 100 DM*

***VIP Professional** plus Desk Assist, ... zusammen nur 850 DM*

***Typesetter Elite** plus Megafont, ... zusammen nur 200 DM*

***Art Director** plus **Film Director**, ... zusammen nur 420 DM*

***Easy Draw II** plus **Multi Hardcopy** ... zusammen nur 450 DM*

1040 ST Color
*plus Art- plus **Film Director**, ... zusammen nur 2650 DM*

wie vor
*plus **OKI 292** (Farbdrucker) plus **Multi Hardcopy**, .. 4500 DM*

Kleines Prolog Lexikon

Adjunktion – Verknüpfung zweier logischer Terme durch **ODER**. Der zugehörige Junktor wird in PROLOG durch das Semikolon ausgedrückt. Beispiel: mensch(X) :– männlich(X);weiblich(X). Ein Mensch ist entweder männlichen oder weiblichen Geschlechtes.

Applikative Sprache – Sprache, in welcher der Algorithmus in den Hintergrund tritt zugunsten der logischen Struktur des Problems. Oftmals auch als loser Oberbegriff für alle deklarativen Sprachen.

Argument – Die Parameter eines Funktors. Z. B. a,b,c in f(a,b,c).

Arität – Anzahl der Argumente eines Funktors. f(a,b,c) hat also die Arität 3.

Atom – Ein konstantes Datensymbol. Muß mit einem Kleinbuchstaben beginnen.

Deklarative Sprachen – Als Gegensatz zu imperativen Sprachen zu verstehen. Man unterscheidet relationale und funktionale Sprachen.

Direktive – Direkt auszuführendes Goal. Beginnt mit :–.

Fakt – Klausel, die nur aus Funktor und Argument besteht. Ein Fakt drückt aus, das der durch Funktor und Argument vorgestellte Zusammenhang wahr ist.

Funktionale Sprachen – Sprachen, bei denen das Ergebnis sich aus der Folge von Funktionen und ihren Argumenten ergibt. Jede Kombination von ARgumenten hat ein eindeutiges Funktionsergebnis.

Funktor – Der Name einer Regel oder eines Faktes. Im Fakt vater(egon) ist vater der Funktor.

Goal – Ziel, die Anfrage, die Prolog veranlaßt, die Datenbasis zu durchforsten.

Imperative Sprachen – Die „klassischen" Sprachen mit dem Schwergewicht auf dem Algorithmusbegriff. Neben der Strukturanalyse für das zu lösende Problem muß sich der Programmierer noch mit dem Weg beschäftigen, wie der Computer die Lösung erarbeiten soll. Jeder einzelne Schritt auf dem Lösungsweg muß dem Computer befohlen werden.

Instanzierung – Bindung einer Variablen an einen Wert (Atom, Liste, Struktur). Nach erfolgter Bindung ist die Variable bis zum Verlassen des Prädikates in dem die Bindung erfolgte nicht mehr rückgängig zu machen. Versucht der Prolog Interpreter per Backtracking neue Instanzierungen vorzunehmen, verläßt er deshalb erst das Prädikat um es durch den **Nochmal** Eingang erneut zu betreten. Danach steht einer erneuten Instanzierung nichts mehr im Wege.

Junktor – Zeichen, welches eine logische Verknüpfung repräsentiert. In PROLOG sind die Junktoren von Negation, Adjunktion und Konjunktion repräsentiert als **not** , und ;.

Konjunktion – Verknüpfung logischer Terme durch **UND**. In PROLOG wird der Junktor als Komma repräsentiert. Beispiel: :– see('datei'),read(X),seen. Eine Direktive zum Lesen der Variable X: Öffne die Datei 'datei' **und** lies X **und** schließe die Datei wieder.

Nebeneffekte – Anweisungen die irgendetwas am Zustand eines Programmes verändern (ohne Neues hinzufügen) haben Nebeneffekte. Beispielsweise ist C eine sehr nebeneffektreiche Sprache. Die Anweisung ++X+=Y–– enthält gleich drei Nebeneffekte. Erst wird X inkrementiert, anschließend Y hinzuaddiert und dann wird Y noch inkrementiert, anschließend Y hinzuaddiert und dann wird Y noch inkrementiert. Auch Prolog enthält Anweisungen mit Nebeneffekten. Beispielsweise bewirkt das Prädikat **write/1** neben dem Erfolg oder Mißerfolg als Nebeneffekt das Erscheinen des Argumentes auf dem Bildschirm (oder in der Datei).

Referentielle Transparenz – Programmiersprachen mit r. T. verhindern, daß eine Variable im Laufe eines Programms verschiedene Werte annimmt. So ist in Prolog eine Variable entweder nicht instanziert oder sie ist instanziert. Im ersten Fall ist die Bindung eines Wertes (oder Struktur) an die Variable möglich. Ist die Variable erst einmal instanziert, ist eine weitere Instanzierung nicht mehr möglich. Da die Instanzierung der Variablen vom Prolog Interpreter vorgenommen wird, ist kein Mißbrauch möglich.

Regel – Klausel, bestehend aus Regelkopf von der Form eines Faktes gefolgt von :– und dem Regelrumpf. Der Regelrumpf besteht aus logischen Verknüpfungen von Fakten und Regeln.

Relationale Sprachen – Prolog ist ein Vertreter dieser Gattung. Die Ausgabe wird in Eigenschaften und Argumenten definiert. Im Gegensatz zu den funktionalen Sprachen sind damit mehrere Ergebnisse pro Prädikat/Argument Paar möglich.

Unifikation – Allgemeiner Mustervergleich. Im Laufe der Unikfikation von Prolog Termen nimmt der Interpreter die Instanzierung von noch nicht instanzierten Variablen vor, um zu einem Erfolg zu kommen. Gelingt es dem Interpreter nicht, die Muster der beiden zu unifizierenden Prologterme in Übereinstimmung zu bringen, meldet der Interpreter einen Mißerfolg. Ist der Lösungsbaum noch nicht abgearbeitet, wird mit Hilfe eines Backtracking Algorithmus nach weiteren Lösungen im Lösungsbaum gesucht. Dabei werden alle Instanzierungen von Variablen die in diesem Unifikationsversuch vorgenommen wurden zurückgenommen. Die entsprechenden Variablen werden im erneuten Unifikationsversuch neu instanziert. Die referentielle Transparenz bleibt also erhalten.

Variable – Ein Objekt, daß mittels Instanzierung an einen Wert gebunden werden kann. Ein Variablenname beginnt immer mit einem Großbuchstaben.

log Interpreter (MPROLOG und das PROLOG des Heim-Verlages), auf die man jetzt schon gespannt sein darf.

Literaturangabe:

---

[1] Clocksin, W. F., C. S. Mellish. Programming in Prolog. Springer Verlag, Berlin 1984.

[2] Coelho, H., J. C. Cotta und L. M. Pereira. How to solve it with Prolog. Laboratorio Nacional de Engenharia Civil, Lissabon 1982.

[3] Colmerauer, A., H. Kanoui, P. Roussel und R. Pasero. Un système de communication homme-machine en français. Groupe d'Intelligence Artificielle, Université d'Aix-Marseille. 1972.

[4] Cuadrado, Clara Y. und John L. Cuadrado. Prolog goes to work. BYTE August 1985, Vol. 10, No. 8. p. 151

[5] Giannesini, F., H. Kanoui, R. Pasero und M. van Ceneghem. PROLOG. Addison Wesley Verlag (Deutschland), 1986.

[6] Sarnow, K. Künstliche Intelligenz – Einführung. ST-Computer 11/86 p. 40.

[7] Sarnow, K. XLISP – Report. ST-Computer 1/87

[8] Schnupp, P. PROLOG Einführung in die Programmierpraxis. Carl Hanser Verlag, München 1986.

[9] Kluzniak, F. und Szpakowicz. Prolog for Programmers. Academic Press, London 1985.





C 64/128
ATARI 520 ST
AMIGA/IBM-PC

PRINT TECHNIK

8000 MÜNCHEN 40
NIKOLAISTR. 2
TEL. 089/368197
TELEX 523203d

SCHWEIZ
MICROTRON
2542 PIETERLEIN·BAHNHOFSTR.2·TEL.032/872429

ÖSTERREICH
PRINT TECHNIK
1060 WIEN · STUMPERGASSE 34 · TEL. 0222/56 4152

EINE NEUE GENERATION VON ST VIDEO DIGITIZERN!

PRO 87 professionell, 128 grau, hohe Auflösung **DM 698,–**

einschließlich Real-Color-Option und Toolbox

PRO 87 ist ein **NEU entwickeltes Produkt** der Hard- und Software-Abteilung der **Print-Technik-München**. Dieses Produkt wurde speziell für den **professionellen und industriellen Markt** entwickelt und zeichnet sich durch die folgenden Auflösungen in **128 Graustufen aus: 320x200, 640x200, 640x400, 512x512, 1024x512**. Die Verwendung von **GEM-Software mit den Drop-Down-Menüs** ermöglicht es dem ST-Besitzer kinderleicht Bilder in den Rechner einzulesen und weiter zu verarbeiten. Die Digitizer-Hardware wird dabei in den **ROM-PORT des ST** eingesteckt und läßt damit den Druckerport für den Drucker frei. Der PRO 87 kann mit dem Videosignal von Kameras (b&w und color), Videorekordern und Fernsehern (Stand-Bild) versorgt werden.
Schwarzweiss (640x400) und Farbmonitor (320x200) werden unterstützt.
Absolut **neu ist die automatische Belichtungsregelung**, erstmals in unseren neuen Digitizern angewandt, was bedeutet, daß man ein optimales Bild ohne vorherige Einstellungen erhält. Will man dennoch Helligkeit und Kontrast ändern, so ist diese Möglichkeit als separater Menüpunkt vorhanden. Jeder dargestellten Graustufe kann man jeweils eine von den **512 Farben** des ST zuordnen. Die Abspeicherung dieser Farbtabellen ermöglicht die Benutzung bei verschiedenen Bildern. Die Bilder können gespiegelt und invertiert werden. Eine spezielle **Zoom**-Option erlaubt es Bildausschnitte vergrößert darzustellen. Das **Histogramm** ermöglicht einen schnellen Überblick über die Graustufenverteilung des Bildes. Die beigelegte **TOOLBOX** ermöglicht das Verkleinern, Biegen, Stauchen, Drehen, etc. von Bildteilen. Abspeicherung der Bilder kann in folgenden Formaten geschehen: DEGAS, NEOCHROM, DOODLE, ART-DIRECTOR und BIT-IMAGE (wird von **DESKTOP**-PUBLISHING Software benutzt). Der **Ausdruck ist in verschiedenen Größen** auf EPSON, ITOH 8510, CANON PJ1080A (Color ink Jet), **Laserdruckern** etc. möglich.

REALTIZER **DM 398,–**

einschließlich Toolbox

Dies ist ein **NEU entwickeltes Produkt** der Hard- und Software-Abteilung der **Print-Technik-München**. Die Verwendung von **GEM-Software mit den Drop-Down-Menüs** ermöglicht es dem ST-Besitzer kinderleicht Bilder in den Rechner einzulesen und weiter zu verarbeiten. Die Digitizer-Hardware wird dabei in den **ROM-PORT des ST** eingesteckt und läßt damit den Druckerport für den Drucker frei. Der Realtizer kann mit dem Videosignal von Kameras (b&w und color), Videorekordern und Fernsehern versorgt werden.
Schwarzweiss (640x400) und Farbmonitor (320x200) werden unterstützt.
Absolut **neu ist die automatische Belichtungsregelung**, erstmals in unseren neuen Digitizern angewandt, was bedeutet, daß man ein optimales Bild ohne vorherige Einstellungen erhält. Will man dennoch Helligkeit und Kontrast ändern, so ist diese Möglichkeit als separater Menüpunkt vorhanden. Die Digitalisierung kann in 2/4/8/16 Graustufen erfolgen (je 1/25 s pro Grau), denen man jeweils eine von den **512 Farben** des ST zuordnen kann. Abspeicherung dieser Farbtabellen ermöglicht die Benutzung dieser bei verschiedenen Bildern. Die Bilder können gespiegelt und invertiert werden. Eine spezielle **Zoom**-Option erlaubt es Bildausschnitte vergrößert darzustellen. Die beigelegte **TOOLBOX** ermöglicht das Verkleinern, Biegen, Stauchen und Drehen von Bildteilen. Abspeicherung der Bilder kann in folgenden Formaten geschehen: DEGAS, NEOCHROM, DOODLE, ART-DIRECTOR und BIT-IMAGE (wird von Desktop-Publishing Software benutzt). Der **Ausdruck ist in verschiedenen Größen** auf EPSON, ITOH 8510, CANON PJ1080A (Color ink Jet), **Laserdruckern** etc. möglich.

Realtizer DM 398,– • Pro 87 Video-Digitizer DM 698,– • Toolbox DM 98,– • Real-Color-Upgrade DM 98,–
Sound Expert-Modul DM 198,– • Sound Master Pro DM 498,– • Speicheroszilloskop DM 498,– • CEBIT 87

Distribution durch Niederlassungen in Europa und Übersee/Nachnahme Versand

---

Aladin

Der Atari ST als Macintosh*-Enhancer

Lassen Sie Ihre Mac-Software auf dem Atari ST laufen

Aladin macht Ihrer Macintosh-Software Beine
- 20% höhere Verarbeitungsgeschwindigkeit
- eine Auflösung von 640*400 Punkten
- eine integrierte, resetsichere, bootfähige Ramdisk

Aladin unterstützt
- Atari ST 260+, 520+ & 1040 mit Monochrome Monitor
- die Laufwerke SF 354 & SF 314

Aladin ist ein deutsches Produkt, der Vorteil für Sie:
- ein deutsches Handbuch
- die deutsche Tastatur wird unterstützt
- eine Hotline direkt zu unserer Entwicklungsabteilung

Großer Public-Domain-Software-Pool
- bis jetzt sind ca. 30 Disketten Macintosh-Software direkt auf Aladin-Format lieferbar

Auf Wunsch liefern wir als Zubehör:
- Kabel: Atari ST - Macintosh, Atari ST - Imagewriter

Aladin kostet:

(Wir nehmen Ihren gebrauchten Mac-Enhancer in Zahlung). Versand per Nachnahme oder Vorauskasse.



zzgl. 4.50 DM Porto & Verpackung

ProficomP

Rappenbergstraße 18a, 7507 Pfinztal 1, Tel: 0721/469229

*Macintosh ist eingetragenes Warenzeichen der Firma Apple Computer.

ATARI ST

Anwendersoftware	
BS-Lager	269,--
BS-Fakt	589,--
BS-Handel	858,--
VIP-Professional dt. Vers.	449,--
dBMAN Version 2.02G	399,--
LASERBASE ST	359,--
K-Spread deutsch	179,--
Music Studio	99,--
Sprachen/Entwicklung/Grafik	
GFA Basic Vers. 2.0	149,--
GFA Compiler	149,--
GFA Vektor 3D Grafik	139,--
GFA Draft CAD Programm	269,--
Metacomco Assembler	159,--
Metacomco Pascal	289,--
Metacomco C-Compiler	319,--
Megamax C-Compiler	449,--
Graphic Artist	649,--
Degas	139,--
Spiele	
Arena	89,--
Borrowed Time	69,--
Brataccas	79,--
Electronic Pool	63,--
Fire Blaster	63,--
Karate Kid II	72,--
Leader Bord Golf	79,--
Little Comp. People	79,--
Silent Service	79,--
Hacker II	79,--
Ultima II	89,--
War Zone	63,--
Winter Games	72,--
World Games	72,--
Super Cycle	69,--

AMIGA

AMIGA 512 KB	nur 2699,--
kompl. mit Maus u. Farbmonitor	
Amiga Druckerkabel	39,--
2 Meter	
Anwendersoftware	
Deluxe Paint	229,--
Deluxe Print	229,--
Deluxe Video	229,--
Music Studio	119,--
Sprachen/Entwicklung/Grafik	
K-Seka Assembler	249,--
Absoft Fortran Compiler	849,--
Absoft Dynamic CAD	1499,--
Metacomo Lisp	499,--
Metacomco Pascal	299,--
Metacomco Toolkit	99,--
TDI Modula II	329,--
Soft Wood Amiga File	329,--
Spiele	
Arena	89,--
Archon II	79,--
Borrowed Time	79,--
Brataccas	89,--
Deep Space	99,--
Leader Bord Golf	72,--
Little Comp. People	89,--
Hacker II	72,--
Marble Madness	79,--
One on One	79,--
Skyfox	79,--
Temple of Asphai	72,--
Winter Games	72,--
World Games	72,--
SS-50 Diskettenbox	nur 24,95
mit Schloß für 50 3,5" Disk	

Thomson Farbmonitor CM 36512 incl. Kabel an ATARI ST	nur 799,--
10er Pack 3,5" Disketten 1 DD 135 tpi	ab 39,--
10er Pack 3,5" Disketten 2 DD 135 tpi	ab 49,--

Wir liefern sämtl. Hard- und Software zu äußerst günstigen Preisen!
Sofort kostenlos Preisliste anfordern!

Computer&Zubehör Versand Gerhard und Bernd Waller GbR
Kieler Str. 689, 2000 Hamburg 54, ☎ 040/5706007 + 5705275

PASCAL ruft TOS

Teil 4 – Eingabe & Ausgabe

In diesem Teil werden einige Routinen besprochen, die eigentlich in jedem Programm benötigt werden. Bestimmt hat sich der eine oder andere schon geärgert, wenn ein Programm in einen Tiefschlaf fällt, weil der angesprochene Drucker nicht eingeschaltet war oder wenn das Programm ständig den gleichen Befehl ausführt, nur weil man einmal zu lange auf eine Taste gedrückt hat und jetzt der Tastaturpuffer unerbittlich arbeitet. Diese Ereignisse können jedoch einfach abgefragt und entsprechend behandelt werden. Oft möchte man in einem Programm die Funktionstasten programmieren oder bestimmte Sondertasten und -kombinationen abfragen und Datum und Uhrzeit neu einstellen. Auch hier bieten die nachfolgenden Routinen einfache Lösungen an.

TASTATUR

Der Tastaturpuffer hat eine Größe von 64 Bytes, d. h. er kann bis zu 64 Zeichen speichern. Wenn also während eines Programmablaufs eine Taste zu lange gedrückt wird, dann übernimmt der Rechner (im Extremfall) diesen Tastencode 64-mal in den Puffer. Ab diesem Zeitpunkt wird bei jeder Funktion, die eine Eingabe des Bedieners erwartet, eines dieser Zeichen aus dem Puffer genommen. In einem Editor konnte es somit passieren, daß dadurch wesentlich mehr Zeilen als gewünscht gelöscht wurden, ohne daß man etwas dagegen tun konnte. Diesem Zustand soll mit der folgenden Routine abgeholfen werden. Dazu wird mit der Funktion:

**function Cconis: integer;
GEMDOS($0B);**

abgefragt, ob sich noch ein Zeichen im Tastaturpuffer befindet. Falls dies der

```
program PUFFER_LOESCHEN;

 var    ch: char;
         i: long_integer;

 function cconis : boolean; gemdos ($0b);
 function crawcin: char;    gemdos ($07);

 procedure LOESCHE_PUFFER;
  var dummy: char;
  begin
    while cconis=true do dummy:=crawcin;
  end;

begin
  loop
    { LOESCHE_PUFFER;       nach erstem Durchlauf aktivieren! }
    writeln('Bitte Taste eine Zeit gedrueckt halten!');
    read(ch);
  exit if ((ch='q') or (ch='e'));
    writeln ('********************',ch);
    i:=0;
    repeat
      i:=i+1;         { Programmschleife }
    until i>100000
  end
end.
```
Listing 1

```
program TASTATUR_REPEAT;

 var a: string;

 procedure Bdrate ( verzoeg, wiederhol: integer ); xbios(35);

begin
  writeln('hohe Wiederholrate:');
  bdrate (50,1); readln(a);
  writeln('niedrige Wiederholrate:');
  bdrate (1,10); readln(a);
  writeln('normal');
  bdrate (10,3); readln(a)
end.
```
Listing 2

```
program CURSOR;

 var ch : char;

 procedure cursor ( funktion, rate: integer ); xbios(21);

begin

  writeln ('Cursor aus');         cursor(0,0);  readln(ch);
  writeln ('Cursor ein');         cursor(1,0);  readln(ch);
  writeln ('Dauercursor');        cursor(3,0);  readln(ch);
  writeln ('Cursor blinkt');      cursor(2,0);  readln(ch);
  writeln ('Cursor langsam');     cursor(4,70); readln(ch);
  writeln ('Cursor schnell');     cursor(4,5);  readln(ch);

end.
```
Listing 3

Fall ist, wird der Wert −1 übergeben, entsprechend der Wert 0, wenn sich kein Zeichen mehr im Puffer befindet.

Um den Puffer zu löschen, wird jeweils ein Zeichen entnommen, und ohne Echo, also nicht auf dem Bildschirm sichtbar, ausgegeben. Dies geschieht mit der Funktion:

function Crawcin: char; GEMDOS($07);

Wenn beide Funktionen in einer Prozedur loesche_puffer zusammengefaßt werden, kann man sie bequem vor jeder Eingabe aufrufen.

Zur Demonstration muß das Programm (Listing 1) zuerst ohne diese Prozedur durchlaufen werden. Dazu wird der Prozeduraufruf einfach als Kommentar geklammert. Wenn jetzt eine Taste längere Zeit gedrückt wird, rotiert das Programm und überspringt die nächsten Eingaben. Wird nun der Prozeduraufruf aktiviert, funktioniert die Eingabe einwandfrei.

Eine andere Möglichkeit, dieses Phänomen zu umgehen, ist die Tastaturwiederholrate hochzusetzen (Listing 2). Dann wird erst nach einem längeren Zeitraum ein weiterer Tastaturdruck registriert und gespeichert:

function Bdrate (start, repeat): integer; XBIOS(35);

Der Parameter START bestimmt, wie lange eine Taste gedrückt werden muß, bis die Wiederholung einsetzt. Als Rückgabewert erhält man die zuvor eingestellten Werte, wobei in den Bits 0–7 der Start- und in den Bits 8–15 der Wiederholwert steht. Wenn für START und REPEAT der Wert −1 angegeben wird, liefert BDRATE nur die momentanen Werte.

CURSOR

Blinkt der Cursor nicht oder zu langsam, dann kann das schnell geändert werden:

procedure Cursor(funktion, wert: integer); XBIOS(21);

Der Parameter WERT ist i. allg. Null; nur bei der Blinkrate hat er eine Bedeutung. Die nachfolgende Tabelle und das Listing 3 zeigen die Möglichkeiten, die sich hier bieten:

```
program FUNKTIONSTASTEN;

 const keyboard = 2;

 var i: integer;

 function bconin( dev: integer ): long_integer; bios(2);

 procedure schluss ; gemdos(0);

 function scan: integer;
   begin
     scan:=int( bconin( keyboard ) div 65535 );
   end;

begin
  writeln('Funktionstastenabfrage (F1-F2)');
  writeln('( Ende mit F10 )');
  writeln;
  while true do begin
    case scan of
      59 : writeln('F1');
      60 : writeln('F2');
      68 : schluss;
      else : writeln( scan )
    end;
  end;
end.
```

Listing 4

```
  program SONDERTASTEN;

   var i,j,k : integer;

   function kbshift: integer; bios(11);

  begin
     repeat
         k:=kbshift;
         case k of
              1: write('-- SHIFT rechts --');
              2: write('-- SHIFT links --');
              4: write('-- CONTROL --');
              8: write('-- ALTERNATE --');
             16: write('-- CAPS LOCK --');
             10: write('-- CONTROL & SHIFT --');
           else: write(k)
         end;
     until k=1
  end.
```

Listing 5

```
Program OUTPUT_READY;

 const drucker = 0;

 function bcostat( dev: integer ): boolean; bios(8);

begin

  if bcostat(drucker) then begin
     rewrite (output,'PRN:');
     writeln ('Hallo, ich bin tatsaechlich angeschaltet!');
     rewrite (output,'CON:');
  end
  else  writeln('Drucker nicht ansprechbar')

end.
```

Listing 6

Funktionsnr.	Bedeutung
0	Cursor abschalten
1	Cursor anschalten
2	Cursor blinkt
3	Cursor blinkt nicht
4	Blinkrate
5	Blinkrate wird übergeben

TASTENABFRAGE

Besonders interessant ist bei vielen Programmen die Abfrage der Funktionstasten. Eine einfache Art ist mit:

function Bconin(dev: integer): long_integer; BIOS(2);

BCONIN wartet auf ein Zeichen von einem Eingabegerät und kehrt erst dann zurück. Der Parameter DEV hat folgende Bedeutung:

0 Centronics Schnittstelle
1 RS232 Schnittstelle
2 Tastatur
3 MIDI Schnittstelle

BCONIN gibt einen entsprechenden Wert zurück, der im Falle einer Tastatureingabe auch den SCAN-Code der gedrückten Taste im Low-Byte des oberen Wortes enthält. Nur dieser Code interessiert für die Abfrage der Funktionstasten, deshalb wird der Rest ausgeblendet. Noch ein Wort zum SCAN-Code: hier werden (fast) alle Tasten des Keyboards einfach durchnummeriert (siehe Grafik 1) und dieser Wert wird an den Rechner übergeben.

Im Beispiel Listing 4 wird beim Drükken der Taste F10 das Programm verlassen. Dies wurde erreicht mit:

procedure Pterm; GEMDOS(0);

Diese Funktion bewirkt einen Rücksprung zum Desktop bzw. zum aufrufenden Programm und stellt deshalb eine gute Abbruchbedingung bei der Funktionstastenprogrammierung dar.

Wie man der SCAN-Code-Tabelle entnehmen kann, existiert dieser nicht für die Tasten CONTROL, SHIFT, ALTERNATE und CAPS LOCK. Diese werden mit einer separaten Funktion abgefragt:

function Kbshift (mode: integer): long_integer; BIOS(11);

Folgende Rückgabewerte sind möglich:

1 rechte SHIFT-Taste
2 linke SHIFT-Taste

```
84 85 86 87 88 89 90 91 92 93   ( geshiftet )
59 60 61 62 63 64 65 66 67 68   ( normal )

1  2  3  4  5  6  7  8  9  10 11 12 13 14 41   98  97      99 100 101 102
15 16 17 18 19 20 21 22 23 24 25 26 27    83   82 72 71    103 104 105 74
   30 31 32 33 34 35 36 37 38 39 40    28 43   75 80 77    106 107 108 78
   96 44 45 46 47 48 49 50 51 52 53                        109 110 112 115
            57                                             113     114
```

```
program UHRZEIT_Setzen;

 procedure GET_ZEIT;

   var zeit,stunden,minuten,sekunden: integer;

   function Tgettime: integer; GEMDOS($2C);

   begin
      zeit:=Tgettime;
      stunden:=shr(zeit,11);
      minuten:=shr((zeit-shl(stunden,11)),5);
      sekunden:=(zeit-shl(stunden,11)-shl(minuten,5));
      writeln(stunden:2,':',minuten:2,':',2*sekunden:2);
   end;

 procedure SET_ZEIT;

   var zeit,stunden,minuten,sekunden: integer;

   procedure Tsettime ( time: integer ); GEMDOS($2d);

   begin
     writeln('STUNDEN MINUTEN SEKUNDEN getrennt durch <Return> eingeben! ')
     read(stunden,minuten,sekunden);
     zeit:=shl(stunden,11)+shl(minuten,5)+sekunden div 2;
     Tsettime( zeit );
   end;

begin

   GET_ZEIT;
   SET_ZEIT;
   GET_ZEIT;
   readln

end.
```

```
program DATUM_Setzen;

 procedure GET_DATUM;
  var datum,tag,monat,jahr: integer;

  function Tgetdate: integer; GEMDOS($2a);

  begin
    datum:=Tgetdate;
    jahr:=shr(datum,9);
    monat:=shr((datum-shl(jahr,9)),5);
    tag:=datum-shl(monat,5)-shl(jahr,9);
    writeln(tag:2,'.',monat:2,'.',jahr+1980:4);
  end;

 procedure SET_DATUM;
  var datum,tag,monat,jahr: integer;

  procedure Tsetdate ( date: integer ); GEMDOS($2b);

  begin
   writeln('TAG MONAT JAHR getrennt durch <Return> eingeben! ');
   read(tag,monat,jahr);
   datum:=shl(jahr-1980,9)+shl(monat,5)+tag;
   Tsetdate( datum );
  end;

begin

  GET_DATUM;
  SET_DATUM;
  GET_DATUM;
  readln

end.
```

4 CONTROL-Taste
8 ALTERNATE-Taste
16 CAPS LOCK-Taste
32 rechte Maustaste oder CLR/HOME
64 linkt Maustaste oder INSERT

Die Zwischenwerte entstehen durch Drücken von Tastenkombinationen, also z. B. CONTROL & SHIFT (siehe auch Listing 5).,

AUSGABEGERÄTE

Wie schon im Vorspann angesprochen, ist es empfehlenswert den Status des Druckers abzufragen, bevor man etwas ausdruckt. Gegebenenfalls kann dann der Benutzer des Programms zum Einschalten aufgefordert werden. Der Aufruf lautet:

function Bcostat(dev: integer): integer; BIOS(8);

Dieser Aufruf dient jedoch nicht nur zur Bereitschaftsabfrage des Druckers, sondern auch die anderen Ausgabegeräte können damit überprüft werden. Dazu muß nur der Parameter DEV entsprechend geändert werden:

DEV-Nr.	Schnittstelle
0	Drucker, Centronics
1	RS232
2	Tastatur
3	MIDI
4	Tastatur-Port

Wenn als Wert eine −1 übergeben wird, dann ist das abgefragte Gerät bereit, bei einer Null ist dies nicht der Fall. Da hier nur die Werte 0 und −1 auftreten, liegt es nahe, den Funktionswert nicht als Integer oder Long_ Integer zu definieren, sondern als BOOLEAN (siehe Listing 6). Die Abfrage wird dann sehr kurz und elegant.

DATUM & ZEIT

Datum und Uhrzeit können sowohl mit zwei getrennten GEMDOS-Funktionen als auch mit einer einzigen XBIOS-Routine gesetzt und verändert werden. Die GEMDOS-Funktionen geben jeweils einen 16-Bit-Wert zurück, die XBIOS-Routine dagegen einen 32-Bit-Wert. Dieser Wert ist wegen seiner Länge etwas unhandlich und soll deshalb hier nicht besprochen werden. Es funktioniert jedoch im Prinzip wie bei den folgenden Programmen (Listing 7 & 8).

Zeit und Datum werden mit den Funktionen:

function Tgettime: integer; GEMDOS($2C);

function Tgetdate: integer; GEMDOS($2A);

erfragt. Sie liegen in der bereits aus Teil 3 bekannten 16-Bit-Form vor, die nun interpretiert werden muß (siehe Listings).

Um Zeit und Datum ändern zu können, müssen sie nach der Eingabe wieder in die 16-Bit-Form gebracht werden. Dazu genügt allerdings ein Einzeiler. Mit dem so erhaltenen Integerwert werden die Funktionen aufgerufen:

procedure Tsettime(time : integer); GEMDOS(★2D);

procedure Tsetdate(date : integer); GEMDOS($2B);

Ein nochmaliges Abfragen der Funktionen oder ein Blick ins Kontrollfeld des Desktop zeigt das neu eingestellte Datum bzw. die Uhrzeit an.

Ich hoffe, daß Sie die vorgestellten Routinen in Ihren Programmen verwenden können und Ihnen somit etwas Mühe erspart wird.

mn

Public-Domain Service

Neu Neu Neu

Ab sofort haben wir unseren Public-Domain-Service umgestellt. Wegen der großen Zahl von Disketten und dem großen Aufwand bei der bisherigen Versandart haben wir uns entschlossen, die Public-Domain-Software mitsamt den Disketten zu liefern. In Zukunft bieten wir die Public-Domain-Software auf sehr guten Disketten zum Unkostenbeitrag von je DM 10,– an.
Bei den Disketten handelt es sich nicht um Noname-Produkte, sondern um qualitativ hochwertige Disketten der Marke **FUJI FILM MF1DD (einseitig 135 TPI)**, mit denen wir gute Erfahrungen sammelten.

Diese Regelung bietet folgende Vorteile:
– mühevolle Verpackung, Formatieren und Beschriften der Disketten, Hin- und Herschicken entfällt.
– einfachere Bearbeitung bei Ihnen und bei uns.
– kürzere Liferzeiten.
– Sie müssen sich nicht um Verpackungsmaterial kümmern, das übernehmen wir.
– und Disketten braucht schließlich jeder, noch dazu zu diesem Preis.
Beachten Sie bitte die Versandbedingungen am Ende der folgenden Seite.

Momentan sind folgende Public-Domain-Disketten vorhanden:

1. Diskette
● *Neochrome V.06:* Malprogramm [f] ● *Doodle:* Malprogramm [s/w]

2. Diskette
● *Joshua:* Diskmonitor – CP/M Emulator (CP/M 80) ● *Megaroids:* Weltraumspiel ● *Diskformatter:* erweitert Diskkapazität auf 399 K bzw. 809 K ● *Kartei-Kasten:* luxuriöses Karteiprogramm [s/w] ● *50/60 Hz*-Umschaltung für Monitore.

Diskette 3 und 4 beinhalten ein komplettes *FORTH*-System *(volksFORTH-83)*. Das System umfaßt einen Screen-Editor, Interpreter, Compiler und diverse Utilities.

5. Diskette
● *Ramdisk* mit Source-Code ● *SG 10* Druckertreiber als Desk-Accessory ● *JAM Term:* Terminalprogramm ● *Diskcopy* ● *ST Grafic:* bewegte Grafiken in 3D ● *CP/M Emulator V 8.4:* neuste Version

6. Diskette
Diese Diskette beinhaltet einige Nutzprogramme für den CP/M-Emulator. Lauffähig mit Version 8.4.
● *CPMTOS:* TOSCPM wandelt CP/M-ASCII-Files in TOS-Format und zurück ● *SCOPY:* Kopierprogramm für ein Laufwerk ● *DDTZ:* Debugger ● *SETKEY:* Tastaturbelegung, u. v. m.!!

7. Diskette
Die Sprache *LISP* (künstliche Intelligenz) ist nun auch auf Public-Domain erhältlich. (incl. 54 kByte Dokumentation, Anleitung und Beispielen)

8. Diskette
● *Diskmon:* ● *GfA Basic-Demo:* ● *Kuvert:* zum Bedrucken von Kuverts ● *Mini-CAD:* ein praktisches Programm mit Source-Code ● *Vokabel:* Vokabeltrainer ● *Superformat:* unterschiedliche Diskformate

9. Diskette
● *Funkplot:* Funktionsplotprogramm incl. Funktionseingabe über Tastatur, Integration und Differenzation ● *Konvert:* Konvertierung von Farbauf Monochrombilder in 3,6 Sekunden ● *Spriteed:* ein kleiner Spriteeditor ● *Super:* ein Superhirnspiel [f]

10. Diskette
● *Iconeditor* ● *Puzzlepuzzle:* Puzzleprogramm ● *Krabat-Schach* Schach für den ST

11. Diskette
● *PROLOG 10* Die Sprache der neuen Generation incl. 95 KByte Anleitung.

12. Diskette
● *Rechner:* ein luxuriöser Taschenrechner ● *Calc:* Taschenrechner als Desk-Accessory ● *Biorhythmus:* Erläuterung überflüssig

13. Diskette
● *Pascalshell:* GEM-Menüsteuerung für ST-Pascal. ● *MMCOPY .ACC:* Filecopy als Accessoir ● *Happy 4:*'Vier gewinnt' ● *Labyrint. ACC:* Labyrint als Accessoir ● *MAD . ACC:* Ihr Rechner entwickelt Eigenleben ● u. a.

14. Diskette
● *Tempelmon:* Monitor von Th. Tempelmann [s/w] ● *Transistorvergleicher:* Transistordatenverwaltung ● *Bitte ein Bit:* Bitcopyprogramm ● *Diskmon* ● *VIFibu:* Kleine Finanzbuchhaltung ● *EX 06:* Fängt Systemfehler und zeigt sie an.

15. Diskette
Druckeranpassung 1st_Word
Auf dieser Diskette befinden sich alle uns zur Verfügung stehenden Druckertreiber. Es sind folgende Treiber enthalten:

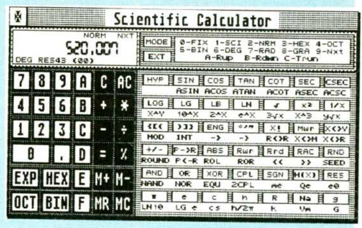
ASCII	CP_80	Epson_12
Epson_LX80	Epson RX80	Gabi 9009
IBM	IBM-Komp	KX-P1092G
LQ_800	MP_165	NEC_P2&P3
NEC_P6&P7	NL10_IBM	NL10_Para
OKI_M182	OKI_M192	QUME
SMM804	TA-SE320	TAXAN 810ST
C 8028	Corona D300	Gabi 9009

16. Diskette
Sammlungen diverser Programme incl. Source Code [s/w] ● *Etikettendruck:* bedruckt Etiketten (Basic) ● *Funkplot:* mit Funktionseingabe (Basic) ● *Kassetten- und Videodatei* unter GEM (Basic) ● *Typewriter:* Schreibmaschinenkurs (Basic) ● *LIFE:* Fortpflanzungs-Rechnung (Pascal) ● *Schiffe:* Schiffe versenken (Demo für GEM-Routinen) (Basic)

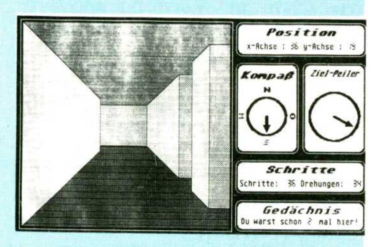
17. Diskette
Programmsammlungsdiskette incl. Source [s/w] ● *Aktien:* Der Aktienmarkt ● *Habu:* Verwaltung von Konten ● *Lohnbuchhaltung & Versanddruck* ● *Steuer:* Einkommens- und Lohnsteuerberechnung.

18. Diskette
● *Doodle plus:* erweiterte Doodle Version; incl. Bilder ● *Power:* Sonderzeichensatz [s/w] ● *Konverter:* Wandelt alle Bildformate ● *Datum:* Datum auf Disk ● *Smooth:* Scrollt eine 64 000 KB Doodle-Bild ● *Format:* erzeugt diverse Formate ● *Kbug:* ST leidet unter Käferbefall

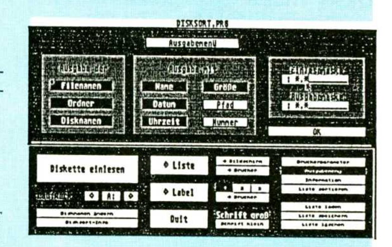
19. Diskette
● *Kermit*plus*:* Stark verbesserte GEM-Version des Terminalprogramms ● *Disksort:* Druckt Diskinhalt als Liste oder als Etikette ● *Terminal:* Terminalprogramm ● *Fontedit:* Zeichensatzeditor für GEM-Draw und Easy-Draw Fonts. [s/w] ● *OKI-Init.ACC:* Steuercodes für Microline 182 incl. Source-Code



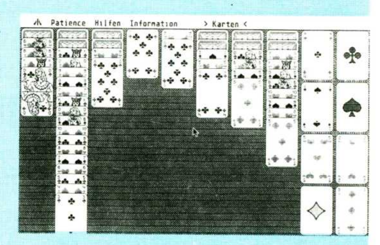
Taschenrechner (PD 12)



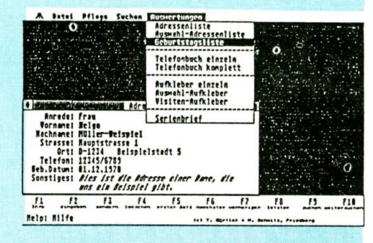
3-D Labyrinth (PD 31)



Disksort (PD 19)

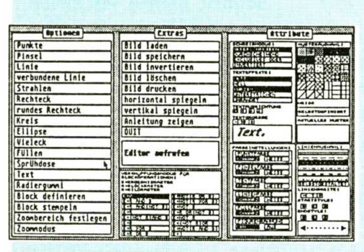


Patience (PD 31)



Adress Pas (PD 29)

ST Public-Domain Service



Andromeda (PD 35)

20. Diskette
● *Harddisk:* Treiber für Harddisk. Automatischer Start der Platte (keine Bootdiskette nötig) ● *Toram:* Kopiert Programme in eine Ramdisk ● *RSCAnalyse:* Analyse von RCS-Files [s/w] ● *DESK Uhr.ACC:* Zeigt ständig Zeit und Datum an. Drei einstellbare Weckzeiten. ● *GEMTOOL:* Demo für GEM-Programmierung [s/w]

21. Diskette
● *Adress:* Private Adressverwaltung ● *Etikett:* Etikettendruck ● *Haushalt:* Privatkontenverwaltung ● *Selektor:* Dateiverwaltung (30 frei erstellbare Felder)

22. Diskette
● *Plot:* Grafische Darstellung von Wertepaaren ● *Myterm:* Terminalprogramm (GfA-Basic) ● *VKT:* Vokabeltrainer

23. Diskette
● *DGDB:* 'Das große deutsche Ballerspiel', Aktionspiel siehe RELAX 12/86 [f] ● *Kalah:* Das bekannte Brettspiel ● *Fractals:* Schnelle 'Apfelmännchen-Version' in Assembler.

24. Diskette
● *Taurus:* Strategiespiel für mehrere Spieler. RELAX 12/86. Schöne Grafik, viele Optionen [f], (leider nur Mega-Rechner)

25. Diskette
● *Disk Engineer:* Diskutility (Rescratch, ...) ● *Omikron Basic:* Run-Only Interpreter des neuen Basics

26. Diskette
● *Codetabelle:* ASCII-Tabelle (Accessoir) [s/w] ● *Life_Game:* Das bekannten Generationenspiel (Accessoir) [s/w] ● *Term:* Und noch ein Terminalprogramm ● *Biorhythmus:* ST-Basic schlägt zurück [s/w] ● *Konto:* Kontoführung in GfA-Basic [s/w] ● *KX P1091:* Accessoir zur Druckersteuerung

27. Diskette
● *NEC-Treiber:*Hardcopy-Treiber für 24-Nadeldrucker (NEC und kompatible). Aufruf über ALT+HELP ● *ST-Sound Demo:* Die Siegerdemo des ST-Computer-Sound-Wettbewerbs. 'Zottl, Knurp und Schnatter hauen rein' (siehe Titelbild ST-Computer Nr. 12)

28. Diskette
● *GfA Basic Programme incl. Source* ● *Funkplot:* [s/w] ● *Terminal:* Terminalprogramm in GfA [s/w] ● *Konverter:* Konvertiert ST-Basic in GfA-Basic ● *ST-Kill:* Utilityprogramme ● *Sprite-Editor:* [s/w]

29. Diskette
● *Diskutility:* Der Weg ans Directory (manipuliert Löschschutz, erzeugt waschechtes IBM-Format, und mehr) [incl. Source in Pascal] ● *Adress_pas:* Adressverwaltung, die einige käufliche Programme das Fürchten lehrt. ● *ST-Graph:* 3-D Balkengrafik [s/w] [incl. Source in GfA]

30. Diskette: Spielesammlung
● *Bohnen* ● *Schiffe:* Schiffe versenken
● *Mastermind:* Superhirn für Superhirne [f]
● *Dragon_hunt:* Ein äußerst gefräßiger Drachen will ausgerechnet Ihnen an den Kragen [f]
● *Memo:* Memory (2 Versionen) [s/w]
● *Solitaire:* Das bekannte 'Taschenspiel' [s/w]
● *4 Gewinnt:* Spiel in GfA-Basic incl. Source [s/w]

31. Diskette: Spielesammlung
● *Patience:* Wunderschöne Version dieses Kartenspiels (siehe Spieletest Nr. 1/87 [s/w] ● *3D-Irrgarten:* Finden Sie einen Weg aus einem dreidimensionalen Labyrinth. Trotz ST-Basic sehr schnell [s/w] ● *Avalun:* Sie verwalten ein Königreich. (ST-Basic)

32. Diskette
● *Mathlib:* Eine umfangreiche Mathematikbibliothek für Pascal. Trigonometrische, Matrizenalgebra und vieles mehr, direkt zum Einbinden
● *Finanzmathe:* Berechnung von Zinsen, Laufzeiten, Laufzeit, Tilgungsplänen, etc.

33. Diskette: diverse Utilities
● *Encrypt:* Ver- und Entschlüsseln von Dateien
● *Ramdisk:* Ramdisk beliebiger Größe ● *Recover:* Regeneriert gelöschte Files (!!) ● *Sterm:* VT52 Terminal + Source in Assembler ● *Freeram:* Zeigt freien Speicher

34. Diskette: Der Assembler zum Kurs
● *Assemblerpaket nach M 68000-Standard* incl. 2-Pass Assembler, Disassembler, Debugger, Editor.

35. Diskette
● *Andromeda:* Zeichenprogramm (DEGAS-ähnlich) ZOOM, Beschriften in vier Richtungen und allen Größen, Kopieren, Linienstil / Art / Breite und alle normalen Grafikbefehle [s/w] ● *3D-Konstruktion:* Zum Entwurf dreidimensionaler Gegenstände [s/w] ● *Globus:* Animierte Erdballgrafik

36. Diskette
● *CLI:* vollständiger Kommandointerpreter (CP/M, MS-DOS Oberfläche) ohne Maus. Ausführen von Batch-Files, Autostart, Kopieren auf Ramdisk, Belegung der Funktionstasten, Ausführen von Programmen (TOS + GEM), etc. ● *IBM. Drive:* Ermöglicht den Filetransfer IBM (--) ST
● *Drupatch:* Zum Ausdruck von Source Files. Mit Zeichenwandlung wie 1ST Word, universell anpaßbar, Perforationssprung. ● *DB_Etikett:* Druckt Etiketten aus DB-Master

37. Diskette
● *SCIENTFIC:* Zeichensatzeditor. Erzeugt Anpassungsfile für 1st Word [s/w] ● *Convert:* Konvertiert ST- und Microsoft Basic Programme in GfA-Basic ● *Check:* Rechtschreibkorrekturprogramm. Erstellt eigenes Lexikon, Übernahme der neuen Wörter, Korrektur direkt im Text.

38. Diskette
Alle Bilder des ST-Kalenderwettbewerbs, die in Heft 1/87 veröffentlicht wurden, incl. Picture-Compactor.

Sonderdisks
Die folgenden Programme sind nicht Public-Domain. Sie können aber bei uns bezogen werden.
A.) – *TOS:* Die letzte Disketten-Version des TOS vom 6.2.1986. Zum fehlerfreien Betrieb des Rechners benötigen Sie diese Version, da z. B. mit älteren Versionen manche Ordner nicht gelesen werden (0 Bytes free).
Unkostenbeitrag 15,– DM incl. Diskette.
B.) – *RCS:* Das Resource Construction Set aus dem ATARI-Entwicklungspaket. Unverzichtbar bei der Erzeugung von Objektbäumen, Dialogboxen, Drop-Down Menüs. Kurzum: Ein Muß für GEM-Programmierung.
Unkostenbeitrag 15,– DM incl. Diskette.
Bis auf die Höhe des Unkostenbeitrags gelten die gleichen Versandbedingungen wie bei der Public-Domain-Software.

Versandbedingungen:

Um einen schnellen und problemlosen Versand zu gewährleisten, beachten Sie bitte folgende Punkte:

– Legen Sie pro bestellter Diskette DM 10,- als Verrechnungsscheck bei (bei Sonderdisks DM 15,-)

– Fügen Sie dem Betrag folgende Versandkosten (Porto + Verpackung) bei:
Inland DM 5,–
Ausland DM 10,–

– Legen Sie unbedingt einen Aufkleber mit Ihrer vollständigen Adresse bei.

– Für ausländische Besteller sei erwähnt, daß Sie mit Euroschecks auch in DM bezahlen können.

– Bitte legen Sie keine zusätzlichen Bestellungen bei (Abo, Zeitschriften, Platinen, etc.), da dies den Vorgang erheblich verlangsamt.

– Außerdem vermerken Sie bitte auf der uns zugesandten Bestellung „Public Domain".

– Zu dem Kostenbeitrag müssen wir leider pro Scheck eine Scheckgebühr von DM 0,50 berechnen. Verwenden Sie deshalb nur einen Scheck, auf dem die Gesamtsumme steht (z. B. für 2 Disketten = DM 20,50).

Auf Wunsch können Sie auch nach den alten Versandbedingungen bestellen, die wir noch zusätzlich aufrechterhalten.

Die Bezugsadresse lautet:

ST-Computer-Redaktion, Postfach 59 69, 6236 Eschborn 1

Das ST-Betriebssystem Teil 8

In diesem achten Teil unserer ST-Betriebssysteme-Serie werden wir die Routinen des XBIOS abschließen. Damit endet auch diese Serie. Man könnte zu diesem Thema unendlich viel schreiben, aber die Serie wurde nur als Anregung gedacht. Spezielle Anwendungen unter TOS werden wir in Zukunft als separate Beiträge bringen.

$ 20 Dosound

Mit Hilfe dieser Funktion ist es möglich, Sounds abarbeiten zu lassen. Die Funktion benötigt als Parameter eine Adresse, die auf eine Tabelle von 'Soundbefehlen' zeigt.

Folgende Befehle stehen dem Anwender zur Verfügung:

Befehl $00–$0F

Dieser Befehl spricht die Registernummern $00–$0F des Soundchips an. Nach der Registernummer folgt ein Byte, das in das Regisdter geladen wird. Die Register des Soundchips sind folgendermaßen definiert:

Register 0
Dient zur Feineinstellung der Frequenz der ersten Stimme.

Register 1
Dient zur Grobeinstellung der Frequenz der ersten Stimme (Hier werden nur die 4 unteren Bit dieses Registers benutzt).

Register 2
Dient zur Feineinstellung der Frequenz der zweiten Stimme.

Register 3
Dient zur Grobeinstellung der Frequenz der zweiten Stimme (Hier werden nur die 4 unteren Bit dieses Registers benutzt).

```
$20 Dosound

move.l #tab,-(sp)  ; Wertetabelle auf Stack
move.w #32,-(sp)   ; Funktionsnummer
trap #14
addq.l #6,sp       ; Stackkorrektur

tab:
dc.b 0,1,1,5,7,254,8,11,1,12,0,7,255
```

```
$21 Setprt

move.w #57,-(sp)   ; Neue Druckerkonfiguration setzen
move.w #33,-(sp)   ; Funktionsnummer
trap #14
addq.l #4,sp       ; Stackkorrektur
```

```
$22 Kbdvbase

move.w #34,-(sp)   ; Funktionsnummer
trap #14
addq.l #2,sp       ; Stackkorrektur
```

```
$23 Kbrate

move.w #1,-(sp)    ; Verzgerungszeit
move.w #1,-(sp)    ; Wiederholungsrate setzen
move.w #35,-(sp)   ; Funktionsnummer
trap #14
addq.l #6,sp       ; Stackkorrektur
```

```
$24 Prtblk

move.l #tab,-(sp)  ; Wertetabelle auf Stack
move.w #36,-(sp)   ; Funktionsnummer
trap #14
addq.l #6,sp       ; Stackkorrektur
tab:
dc.l $f8000,0,50,50,100,100,2,0,1,1,1
```

```
$25 Vsync

move.w #36,-(sp)   ; Funktionsnummer
trap #14
addq.l #2,sp       ; Stackkorrektur
```

Register 4
Dient zur Feineinstellung der Frequenz der dritten Stimme.

Register 5
Dient zur Grobeinstellung der Frequenz der dritten Stimme (Hier werden nur die 4 unteren Bit dieses Registers benutzt).

Register 6
Dient mit seinen letzten fünf Bit zur Einstellung der Rauschfrequenz.

Register 7
Multifunktionsregister

Bit 0: Ist dieses Bit nicht gesetzt, so wird die erste Stimme eingeschaltet. Ist das Bit gesetzt, ist sie ausgeschaltet.

Bit 1: Ist dieses Bit nicht gesetzt, so wird die zweite Stimme eingeschaltet. Ist es gesetzt, ist sie ausgeschaltet.

Bit 2: Ist dieses Bit nicht gesetzt, so wird die dritte Stimme eingeschaltet. Ist es gesetzt, ist sie ausgeschaltet.

Bit 3 und 4 dienen dazu den Rauschgenerator für alle drei Stimmen ein- oder auszuschalten. Auch hier bedeutet ein gesetztes Bit das Ausschalten des Rauschgenerators und umgekehrt.

Bit 6 und 7 steuern Port A bzw. Port B.

Register 8
Durch dieses Register (untere vier Bits) wird die Lautstärke der ersten Stimme geregelt.

Register 9
Genauso wie Register 8, aber für die zweite Stimme.

Register 10
Wie Register 8 und 9, aber für die dritte Stimme.

Register 11 und 12
Durch diese 16 Bits wird die Periodendauer der Kurvenform der Hüllkurve beeinflußt.

Register 13
Bestimmt die Form der Hüllkurve.

Register 14 und 15
Dieses Register dient zum Ansprechen der zwei Ports dieses Bausteins.

Befehl $80

Durch diesen Befehlstyp wird das nachfolgende Argument (ein Byte) in ein temporäres Register geladen.

Befehl $81

Bei diesem Befehl werden drei Argumente benötigt. Das erste Argument ist die Nummer des Registers des Soundchips, in das der Inhalt des temporären Registers übergeben werden soll. Das zweite Argument ist das Zweierkomplement des Inhalts des temporären Registers, das zu diesem Register addiert wird. Das dritte Argument enthält den sogenannten Endwert. Der Befehl wird so lange ausgeführt, bis der Endwert erreicht ist.

Befehl $82 – $FF

Durch diesen Befehl wird die Länge eines Sounds bestimmt. Ist das Argument eine Null, so wird der Sound unterbrochen.

$21 Setprt

Durch diese Funktion läßt sich die Druckerkonfiguration abfragen oder neu einstellen. Wird als Parameter eine –1 übergeben, so erhält man in D0 die aktuelle Konfiguration zurück. Möchte man den Drucker neu konfigurieren, so wird dem Stack ein Wert übergeben, der dieser neuen Konfiguration entspricht.

Hier die Bedeutung aller 16 Bit dieses Langworts:

Bit 0
Matrixdrucker 0 / Typenraddrucker 1

Bit 1
Farbdrucker 0 / Monochromdrucker 1

Bit 2
1280 Dots 0 / 960 Dots 1

Bit 3
Test-Modus 0 / Maximum-Modus 1

Bit 4
Drucker 0 / Modem 1

Bit 5
Endlospapier 0 / Einzelblatt 1

Bit 6–14
reserviert

Bit 15
immer 0

$22 Kbdvbase

Übergibt die Anfangsadresse einer Tabelle, in der Adressen von Routinen stehen, die Daten des Tastaturprozessors verarbeiten. Die Tabelle ist folgendermaßen aufgebaut:

1. Long-Word MIDI Eingabe
2. Long-Word Tastatur Fehler
3. Long-Word MIDI Fehler
4. Long-Word IKBD Status
5. Long-Word Maus-Routinen
6. Long-Word Uhrzeit-Routine
7. Long-Word Joystick-Routine

$23 Kbrate

Durch diese Funktion kann man die Wiederholungsrate der Tastatur ermitteln oder neu setzen. Übergibt man als Parameter eine –1, so wird der alte Wert zurückgegeben. Um eine neue Wiederholungsrate einzustellen, müssen zwei Parameter dem Stack übergeben werden. Der erste Parameter gibt die Verzögerungszeit nach einem Tastendruck an, die bestimmt, ab wann die automatische Wiederholsequenz startet. Der zweite Parameter dient zur Definition des Geschwindigkeitsfaktors.

$24 Prtblk

Gibt eine definierte Hardcopy auf dem Drucker aus. Diese Funktion ist der Scrdmp-Funktion (20) sehr ähnlich und wird von dieser auch benutzt. Die Funktion erwartet wieder eine Anzahl von Parametern. Die Adresse, die auf die Parameterliste zeigt, wird dem Stack weitergegeben.

Die Parameter-Liste ist folgendermaßen aufgebaut:

Parameter	Typ	Bedeutung
1.	Long-Word	Adresse des Bildschirms
2.	Integer	Null
3.	Integer	Breite des Bildschirms
4.	Integer	Höhe des Bildschirms
5.	Integer	Linker Rand
6.	Integer	Rechter Rand
7.	Integer	Auflösung des Bildschirms
8.	Integer	Auflösung der Drucker
9.	Integer	Zeiger für Farbpalette
10.	Integer	Druckertyp
11.	Integer	Druckerport (Centronics, RS 232)
12.	Long Word	Zeiger für Halbtonmaske

$25 Vsync

Nach Aufruf dieser Funktion wird auf den nächsten Vertikal-Blank-Interrupt gewartet. Anschließend erfolgt ein sofortiger Rücksprung. Mit dieser Funktion ist es zum Beispiel möglich, Grafik-Operationen synchron laufen zu lassen.

$26 Supexec

Läßt eine Routine im Supervisor-Modus ablaufen. Diese Funktion ist der GEMDOS-Funktion Super ($20) sehr ähnlich. Der große Vorteil von Supexec ist, daß nicht das ganze System in den Supervisor-Modus versetzt wird. Man übergibt die Anfangsadresse einer Routine, die im Supervisor-Modus ausgeführt werden soll. Man sollte darauf achten, daß die Routinen keine BIOS- oder GEMDOS-Funktionen enthalten.

$27 Puntaes.

Schaltet das AES ab. Der von AES belegte Speicherplatz wird freigegeben.

```
$26 Supexec

move.l #routine,-(sp)  ; auszufhrende Routine auf Stack
move.w #38,-(sp)       ; Funktionsnummer
trap #14
addq.l #6,sp           ; Stackkorrektur
clr-(sp)               ; GEMDOS 0
trap #1
routine:               ; Hardcopy-Routine
move.l #1,$4ee
rts
```

```
$27 Puntaes

move.w #39,-(sp)       ; Funktionsnummer
trap #14
addq.l #2,sp           ; Stackkorrektur
```

Dialogboxen in GFA Basic

Dies ist eine Beispielbox
ST Computer

Taste 1   Taste 2

EDIT-Feld: Dialog in GFA

Mit dem GFA Basic Interpreter ist es leicht möglich, einige GEM-Elemente wie z. B. Alert-Boxen zu programmieren, da die nötigen Befehle bereits implementiert sind. Nun stellt sich einem das Problem, daß diese Alert-Boxen nur für irgendwelche Bildschirmmeldungen geeignet sind. Um nun auch Eingaben in Boxenform realisieren zu können, muß man sogenannte Dialogboxen kreieren. Dies ist für GFA Basic mittels ein paar AES-Routinen und einem Resource Construction Set einfach zu bewerkstelligen.

Wenn Sie jetzt denken, daß ist ja alles schön und gut, nur ich habe kein Resource Construction Set, also höre ich gleich auf zu lesen, dann lesen Sie bitte doch noch einen Satz weiter und entscheiden Sie dann. Die ST Computer Redaktion bietet ab der Februar-Ausgabe das **Resource Construction Set** aus dem ATARI Entwicklungspaket **für DM 15,– inkl. Diskette** an. Es sei betont, daß diese Diskette keine Public Domain ist und somit nicht frei kopierbar ist. Das Resource Construction Set sollte also nicht das Problem sein.

Doch nun zurück zum Thema Dialogobxen. Wir wollen nicht allzu tief in die Materie von Objektbäumen, TEDINFO, u. ä. Dingen eingehen, da diese bereits in unserem GEM-Kurs Teil 2 Abschnitt 2.2 in der Januar-Ausgabe behandelt wurden und auch im Rahmen dieses Kurses weiter besprochen werden. Dieser Artikel soll also nur zeigen, wie man mittels einem Resource Construction Set erstellte Dialogboxen in das GFA Basic einbinden kann.

Benutzt wird in diesem Artikel das GFA Basic V2.0, da hiermit eine Parameterrückgabe aus Prozeduren leicht zu bewerkstelligen ist. Es sei sowieso jedem GFA Basic Besitzer empfohlen, den Update-Service auszunutzen. Es rentiert sich.

```
' ****************************
' *                          *
' *  RSC-Dialog in GFA-Basic  *
' *                          *
' ****************************
'
'
Gosub Rsrc_load("Beispiel.rsc",*Fehler%)
If Fehler%=0 Then
  Print "RSC-Ladefehler"
  End
Endif
'
Gtype%=0
Gosub Rsrc_gaddr(Gtype%,Gindex%,*Baum%,*Fehler%)
Gtype%=2
Gosub Rsrc_gaddr(Gtype%,Gindex%,*Tedinfo%,*Fehler%)
'
Gosub Form_center(Baum%,*Fo_cx%,*Fo_cy%,*Fo_cw%,*Fo_ch%)
Gosub Form_dial(0,0,0,0,0,Fo_cx%,Fo_cy%,Fo_cw%,Fo_ch%,*Fehler%)
Gosub Form_dial(1,0,0,64,32,Fo_cx%,Fo_cy%,Fo_cw%,Fo_ch%,*Fehler%)
Gosub Objc_draw(Baum%,0,1,Fo_cx%,Fo_cy%,Fo_cw%,Fo_ch%,*Fehler%)
Gosub Form_do(Baum%,4,*Ex_obj%)
If Ex_obj%=2 Then
  Print "Taste 1 wurde gedrückt !"
Else
  Print "Taste 2 wurde gedrückt !"
Endif
Gosub Form_dial(2,0,0,64,32,Fo_cx%,Fo_cy%,Fo_cw%,Fo_ch%,*Fehler%)
Gosub Form_dial(3,0,0,0,0,Fo_cx%,Fo_cy%,Fo_cw%,Fo_ch%,*Fehler%)
Gosub Get_dialogtext(*Txt$)
Print Txt$
'
Gosub Rsrc_free(*Fehler%)
End
'
'
Procedure Rsrc_load(Rscname$,Aes_return%)
  '
  Lpoke Addrin,Varptr(Rscname$)
  Gemsys 110
  *Aes_return%=Dpeek(Gintout)
Return
'
Procedure Rsrc_gaddr(Type%,Index%,Aes_adr%,Aes_return%)
  '
  Dpoke Gintin,Type%
  Dpoke Gintin+2,Index%
  Gemsys 112
  *Aes_return%=Dpeek(Gintout)
  *Aes_adr%=Lpeek(Addrout)
Return
'
Procedure Form_center(Baum%,F_cx%,F_cy%,F_cw%,F_ch%)
  '
  Lpoke Addrin,Baum%
  Gemsys 54
  *F_cx%=Dpeek(Gintout+2)
  *F_cy%=Dpeek(Gintout+4)
  *F_cw%=Dpeek(Gintout+6)
  *F_ch%=Dpeek(Gintout+8)
Return
'
Procedure Form_dial(Flag%,Littlx%,Littly%,Littlw%,Littlh%,Bigx%,Bigy%,Bigw%,Bigh%
return%)
  '
  Dpoke Gintin,Flag%
  Dpoke Gintin+2,Littlx%
  Dpoke Gintin+4,Littly%
  Dpoke Gintin+6,Littlw%
  Dpoke Gintin+8,Littlh%
```

AES heißt das Zauberwort

Mittels einiger AES-Routinen kann man das nötige Resource-File einladen, die Box erscheinen lassen und anschließend den Texte eingeben und weiterverwerten. Für jede AES-Routine wird eine eigene Prozedur benutzt, so daß man diese auch für andere Programme problemlos nutzen kann. Die Routinen müssen in folgender Reihenfolge angesprochen werden:

Rsrc_load

Hiermit wird Speicherplatz für das Resource-File reserviert und es anschließend in den Speicher geladen. Ist der Rückgabeparameter = 0, dann ist ein Fehler beim Laden aufgetreten. Als Paramet er wird der Resource-Filename benötigt.

Rsrc_gaddr

Diese Funktion dient zur Ermittlung der Anfangsadresse des Resource-Files (Gtype%=0) und der TEDINFO (Gtype%=2). Die wichtigen Parameter sind **Gtype%**, der bestimmt, welche Adresse ermittelt werden soll, und **Baum%** bzw. **Tedinfo%**, die nach Aufruf der Routine die Adresse des Objektbaumes des Resource-Files bzw. der TEDINFO im Speicher enthält.

Form_center

Nach Aufruf dieser Routine bekommt man die Koordinaten der auf dem Bildschirm mittig zentrierten Dialogbox als Rückgabeparameter (X- und Y-Koordinate, Breite und Höhe der Box). Form_center benötigt nur die Adresse **Baum%** vor dem Aufruf.

Form_dial

Form_dial wird gleich viermal aufgerufen (zweimal vor den Objc_draw- und Form_do-Routinen und zweimal dahinter). Das kommt daher, da diese Routine im Grunde genommen aus vier Unterroutinen (0–3) besteht. Diese haben folgende Funktionen:

0 = reserviert einen Bildschirmbereich für die Box
1 = zeichnet ein sich vergrößerndes Rechteck
2 = zeichnet ein sich verkleinerndes Rechteck
3 = gibt den reservierten Bildschirmbereich wieder frei

```
    Dpoke Gintin+10,Bigx%
    Dpoke Gintin+12,Bigy%
    Dpoke Gintin+14,Bigw%
    Dpoke Gintin+16,Bigh%
    Gemsys 51
    *Aes_return%=Dpeek(Gintout)
    If Flag%=3
      Put Bigx%,Bigy%,Rette$
    Endif
  Return
  '
  Procedure Objc_draw(Tree%,Startob%,Depth%,Xclip%,Yclip%,Wclip%,Hclip%,Aes_return%
    '
    Get Xclip%,Yclip%,Xclip%+Wclip%+1,Yclip%+Hclip%+1,Rette$
    Dpoke Gintin,Startob%
    Dpoke Gintin+2,Depth%
    Dpoke Gintin+4,Xclip%
    Dpoke Gintin+6,Yclip%
    Dpoke Gintin+8,Wclip%
    Dpoke Gintin+10,Hclip%
    Lpoke Addrin,Tree%
    Gemsys 42
    *Aes_return%=Dpeek(Gintout)
  Return
  '
  Procedure Form_do(Tree,Startob%,Aes_return%)
    '
    Dpoke Gintin,Startob%
    Lpoke Addrin,Tree
    Gemsys 50
    *Aes_return%=Dpeek(Gintout)
  Return
  '
  Procedure Get_dialogtext(Txt%)
    '
    Txt_adr%=Lpeek(Tedinfo%)
    Te_txtlen%=Dpeek(Tedinfo%+24)-1
    For I=0 To Te_txtlen%
      Zeichen=Peek(Txt_adr%+I)
      If Zeichen=0 Then
        I=Te_txtlen%
      Endif
      If Chr$(Zeichen)<>"_" Then
        Txt$=Txt$+Chr$(Zeichen)
      Endif
    Next I
    *Txt%=Txt$
  Return
  '
  Procedure Rsrc_free(Aes_return%)
    '
    Gemsys 111
    *Aes_return%=Dpeek(Gintout)
  Return
```

Die Parameter für den Form_dial-Aufruf lauten:

Flag% = Funktionsnummer (0–3)
Littlx% = X-Koordinate des Rechtecks in der kleinsten Form
Littly% = Y-Koordinate des Rechtecks in der kleinsten Form
Littlw% = Breite des Rechtecks in der kleinsten Form
Littlh% = Höhe des Rechtecks in der kleinsten Form
Bigx% = X-Koordinate des Rechtecks in der größten Form
Bigy% = Y-Koordinate des Rechtecks in der größten Form
Bigw% = Breite des Rechtecks in der größten Form
Bigh% = Höhe des Rechtecks in der größten Form

Jetzt kommt vielleicht die Frage auf, welches Rechteck und wieso kleinste bzw. größte Form. Das ist einfach zu beantworten. Die Form_dial-Unterroutine 1 zeichnet ein sich vergrößerndes Rechteck und benötigt dazu die Littl-Parameter als Ursprungsgröße und die Big-Parameter als Endgröße des Rechtecks. Mit anderen Worten das Rechteck wächst von der linken oberen Ecke (X=0, Y=0) mit einer Breite von 64 und einer Höhe von 32 in die Bildschirmmitte. Bei der Unterroutine 2 verhält es sich genau umgekehrt, da das Rechteck ja von der maximalen Größe auf die kleinste schrumpfen soll.

Objc_draw

Diese Routine zeichnet die Dialogbox und benötigt folgende Parameter:

Baum% = Adresse des Objektbaumes

Startob% = Index des ersten zu zeichnenden Objektes (im Bsp. 0)

Depth% = Anzahl der Ebenen, die zu zeichnen sind (im Bsp. 1)

Xclip%, Yclip%, Wclip% und Hclip% = Koordinaten der Dialogbox

AES erlaubt maximal neun Ebenen (0–8), aus denen sich eine Box zusammensetzen kann. In unserem Falle werden allerdings nur die Ebenen 0 und 1 benötigt.

Form_do

Diese Routine übergibt die weitere Kontrolle des Programms an das AES und wartet auf Eingaben in der Dialogbox. Es werden folgende Parameter für den Aufruf benutzt:

Baum% = Adresse des Objektbaumes

Startob% = Index des editierbaren Objektes (im Bsp. 4)

Als Rückgabeparameter erhält man in Ex_obj% den Index des Objektes (siehe Tabelle), das man in der Box angeklickt hat. In unserem Beispiel wären nur die Objekte 2 (Taste 1) und 3 (Taste 2) möglich, da diese als Ausgangsbuttons definiert wurden.

Tabelle der Objekte

Objekt 0 = Dies ist eine Beispielbox
Objekt 1 = ST Computer
Objekt 2 = Taste 1
Objekt 3 = Taste 2
Objekt 4 = EDIT-Feld:__________

Form_dial

Siehe oben

Get_dialogtext

Diese Prozedur ist kein AES-Routinenaufruf, sondern liest nur den Text der Eingabe Zeichen für Zeichen aus dem Speicher. Die Adresse erhält man durch das Auslesen einer Adresse der TEDINFO (Txt_adr%). Auf dieselbe Art und Weise bekommt man die Länge des editierten Textes (Te_txtlen%). Es würde jetzt zu weit führen den kompletten Aufbau der TEDINFO zu erklären, so daß hier diese zwei Adressen genügen sollen.

Rsrc_free

Jedes Laden eines Resource-Files reserviert und verbraucht somit auch logischerweise Speicherplatz. Beim Austesten seiner Programme sollte man deswegen die Funktion Rsrc_free vor dem erneuten Start aufrufen, da ansonsten irgendwann kein Speicherplatz mehr vorhanden ist. Natürlich empfiehlt sich der Aufruf auch bei Beendigung des Programms.

Ein Loch auf dem Bildschirm

In manchen Programmen hinterläßt der Aufruf von Dialog- und Alertboxen häßliche Löcher auf dem Bildschirmaufbau. Das kommt daher, daß die Programmierer versäumt haben vor dem Objc_draw-Aufruf den Bildschirminhalt zu retten, der durch die Boxen überlagert wird. In GFA Basic ist dies ganz einfach zu bewerkstelligen. Man benutzt einfach den GET-Befehl, um den Bildschirmteil in einem String abzuspeichern. Nach dem Schließen der Box, wird dann der Bildschirmteil wieder mittels des PUT-Befehls ergänzt, so daß kein Loch zurückbleibt.

Wir hoffen, daß das kleine Beispiel zeigt, wie man sich mit einfachen Mitteln Dialogboxen für eigene Anwendungen erstellen kann und verweisen nochmals für genauere Informationen über die Objektbaumstrukturen auf den GEM-Kurs.

(HE / Stefan Höhn)

Formelberechnung mit GfA-BASIC

Oft findet man in Computerzeitschriften und -büchern Programme in allen möglichen Sprachen, die Berechnungen oder graphische Darstellungen mit mathematischen Funktionen durchführen. Meist sind jedoch die Formeln fest im Programm „eingebaut", und Änderungen können zum Teil mit erheblichem Aufwand verbunden sein, insbesondere wenn man mit Compilern arbeitet.

Wer mit einem BASIC-Interpreter herkömmlicher Art arbeitet, der über die Befehle CHAIN und MERGE verfügt, kann sich mit einem Trick aus der Patsche helfen. Wie in Listing 1 gezeigt ist es möglich, über den Umweg einer Diskettendatei ein sich selbst veränderndes Programm zu schreiben.

Programmiert man jedoch in einer Compilersprache oder z. B. mit dem enorm leistungsfähigen GfA-BASIC-Interpreter, kommt man mit solchen Kniffen nicht mehr weiter.

Hier hilft alles nichts, man muß sich sozusagen zu Fuß durch eine als Zeichenkette eingegebene Formel „beißen" und Zeichen für Zeichen analysieren.

So aufwendig, wie das auf den ersten Blick scheinen mag, ist dies nun auch wieder nicht; das Zauberwort, das hier weiterhilft heißt Rekursion.

Schauen wir uns doch einmal eine Formel – einen arithmetischen Ausdruck wie die Mathematiker sagen – genauer an. Zur graphischen Verdeutlichung sollen hier sogenannte Syntax-Diagramme, ähnlich wie man sie zur Definition von PASCAL-Befehlen verwendet, weiterhelfen. Bild 1 erklärt die möglichen Symbole. Syntax-Diagramme werden in Pfeilrichtung auf Flußlinien durchlaufen; diese können sich an verschiedenen Stellen aufspalten und so Möglichkeiten zur Verzweigung und Schleifenbildung beschreiben. Die Kästchen mit den abgerundeten Ecken dienen nur der Namensgebung des jeweiligen Diagramms und können einfach übergangen werden.

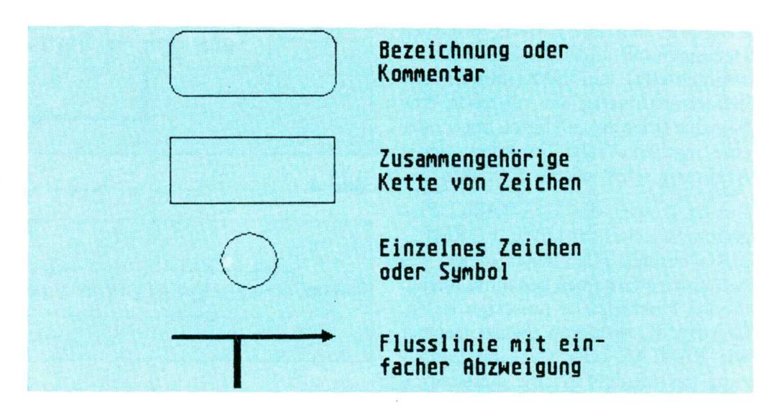


Bild 1: Definition der für die Syntax-Diagramme verwendeten Symbole

Unsere Formel kann nun als Kette von Zeichen, die sich in bestimmte Teilketten zerlegen läßt, mit diesem Schema formal beschrieben werden. So, wie man den arithmetischen Ausdruck von links nach rechts zu Papier bringt, kann man den Flußlinien folgen und, gemäß der mathematischen Hierarchie der Rechenregeln, von einer Grobdarstellung zu immer weiteren Unterteilungen gelangen.

Wohlgemerkt, Syntax-Diagramme sind weder Flußdiagramme noch dienen sie zur exakten Darstellung eines Programmablaufs, jedoch lassen sich die durch sie beschriebenen Regeln leicht in entsprechende Programm-Algorithmen umsetzen.

In Bild 2 ist nun allgemein ein arithmetischer Ausdruck dargetellt, der aus beliebig vielen einfachen Ausdrücken besteht, die über das Plus- oder Minuszeichen verknüpft sein können. Dies ist durch die Abzweigung hinter dem Kasen „Einfacher Ausdruck" verdeutlicht, was bedeutet, daß hier abgebogen wird solange einem Teilausdruck ein Plus- oder Minusoperator folgt.

Beispiel:

6.89 + (3/7.8 – 5)*5 – sin(exp(– 3*7/8)) + ...

6.89, (3/7.8 – 5)*5, sin(exp(– 3*7/8)) sind Teilausdrücke des o. g. arithmetischen Audrucks

Jeder einfache Ausdruck setzt sich aus einzelnen Termen zusammen, die über Multiplikation oder Division verknüpft sind (Bild 3).

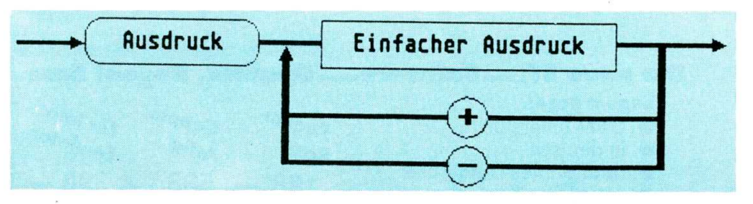


Bild 2

Ein Term ist ein vorzeichenbehafteter Faktor, der gegebenenfalls wiederum mit einem vorzeichenbehafteten Faktor potenziert sein kann (Bild 4).

Ein vorzeichenbehafteter Faktor besteht aus einem vorzeichenlosen (positiven) Faktor mit eventuell vorangestelltem negativem Vorzeichen (Bild 5).

Wie in Bild 6 zu sehen, ist nun ein Faktor entweder eine positive reelle Zahl (die sich aus Ziffern, eventuell Dezimalpunkt und/oder Exponent zusammensetzt), ein in Klammern stehender arithmetischer Ausdruck, eine Variable (hier als „x" bezeichnet) oder eine mathematische Funktion, deren Argument selbst wieder ein Faktor ist.

Listing 2 zeigt ein GfA-BASIC Programm, welches das Modul FORMEL.MRG enthält. Hier wird gezeigt, wie sich konsequent oben genannte Regeln ine ine Hochsprache umsetzen lassen. GfA-BASIC ist hierzu ebenso geeignet wie PASCAL oder C. Eine Umsetzung des Moduls in eine dieser Sprachen sollte daher keine Probleme bereiten. Bei FORMEL.MRG handelt es sich um ein universell benutzbares Teilprogramm, das leicht an unterschiedliche Anwendungen angepaßt werden kann.

Das eigentliche Hauptprogramm mit der Procedure Def_funktion dient nur der Demonstration von FORMEL.MRG. Das hier vorgestellte Modul kann sehr vielseitig verwendet werden, z. B. für einen selbstprogrammierten Taschenrechner, ein Funktionsplotprogramm oder die Analyse von mathematischen Funktionen. Erweiterungen mit beliebig komplexen Funktionen sind leicht durchzuführen, wie im Programmtext an entsprechender Stelle kommentiert wurde.

Ich hoffe, FORMEL.MRG gibt viele Anregungen für eigene Ideen und wünsche viel Spaß beim Programmieren.

(H. Bauch)

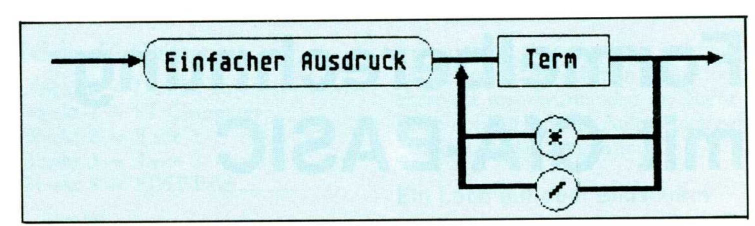


Bild 3

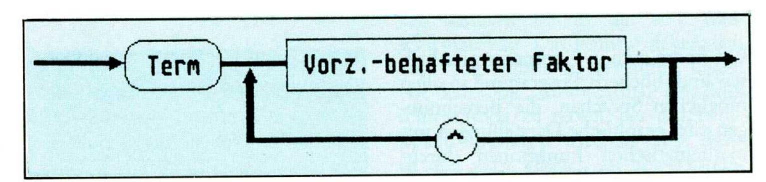


Bild 4

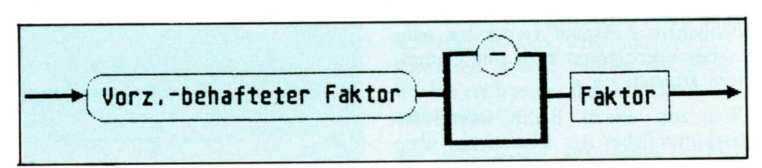


Bild 5

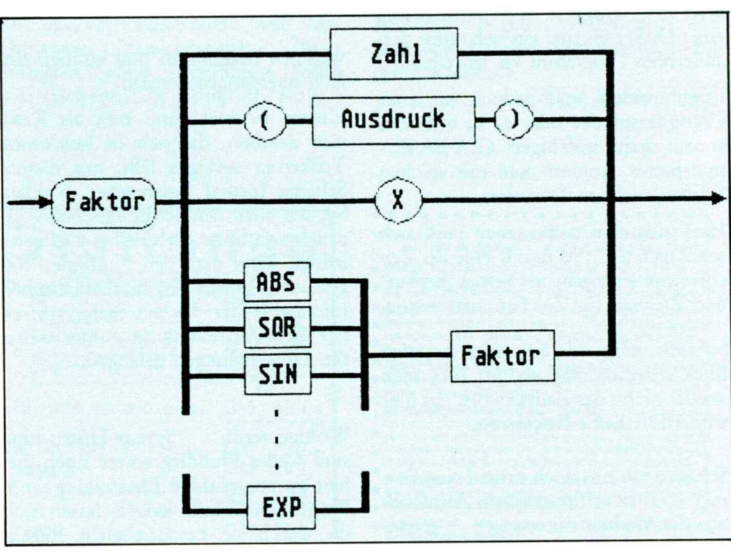


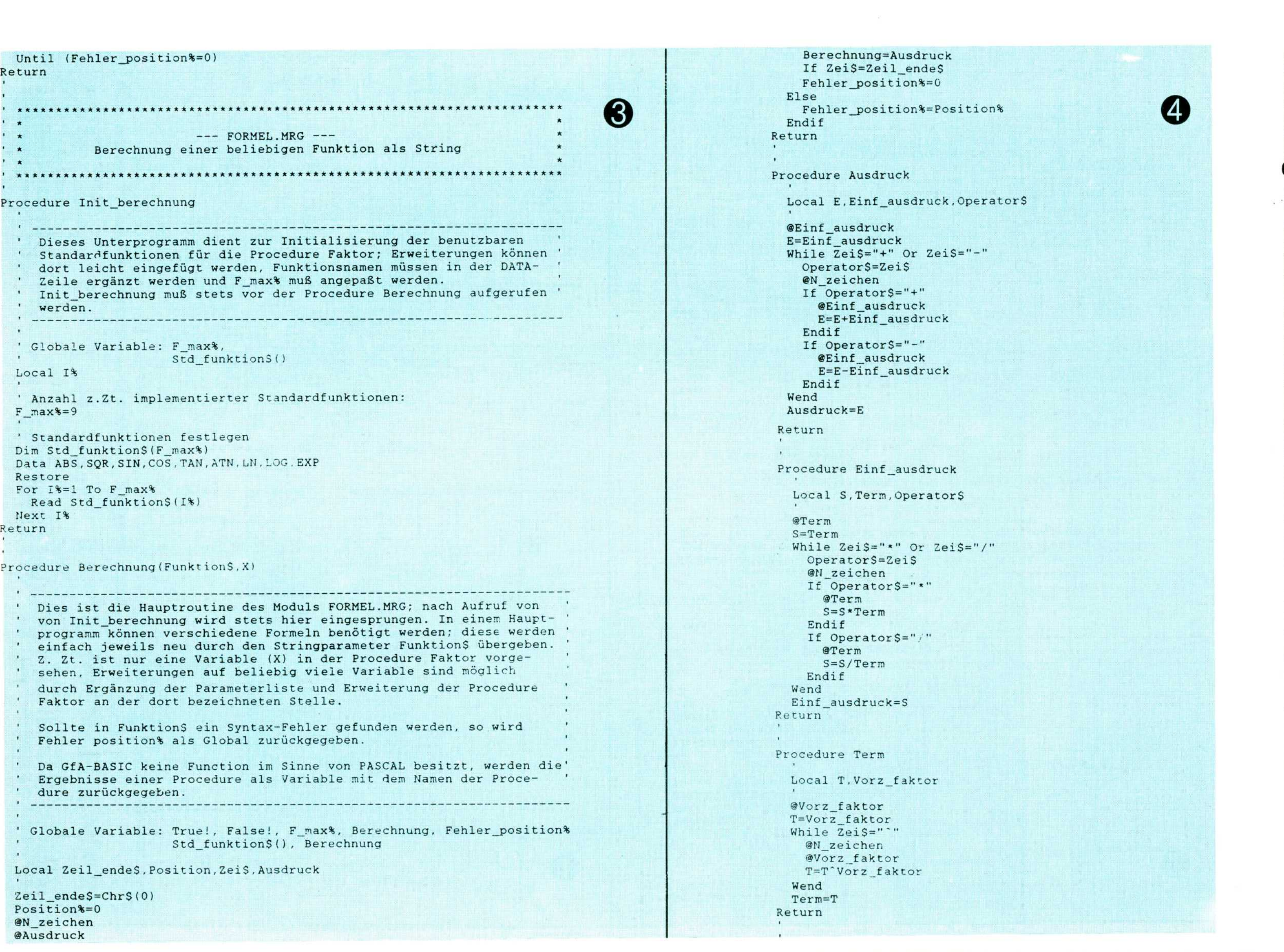
Bild 6

1

```
'**********************************************************
'*                                                        *
'*                --- ST_FORM.BAS ---                     *
'*       Beispiel zur Berechnung beliebiger Formeln       *
'*             mit dem ST-BASIC-Interpreter               *
'*                                                        *
'*          (c) 22.11.1986     H. Bauch                   *
'*                                                        *
'**********************************************************
Clearw 2:Fullw 2
Print
Print "            F O R M E L B E R E C H N U N G "
Print "           ___________________________________"
Print
gosub Funktion.def
'
Anfang:
Input " x = ",x$
If x$<>"N" and x$<>"n" Then
Print " f(x) = ":fny(val(x$)),"<N> für neue Funktion !"
Else Gosub Funktion.def
Goto Anfang
'
'
Funktion.def:
Print
Print " Definition einer beliebigen Funktion in BASIC-Syntax"
Print
Input " f(x) = ",Formel$
Open "O",#1,"FKT.BAS"
Print#1,"200 Def fny(x) = "+Formel$
Close #1
Chain Merge "FKT.BAS",200
' Die nächste Zeile wird vom Programm verändert:
Def fny(x) = 1/(sqr(1-x^2))
Return
'******************************************************************
' *                                                               *
' *                    --- FORMULA.BAS ---                        *
' *           Ein Testprogramm für das Modul FORMEL.MRG           *
' *                                                               *
' *              (c) 23.11.1986  H. Bauch                         *
' *                              Taunusstr. 96                    *
' *                              6237 Liederbach                  *
' *                                                               *
'******************************************************************
'
' --- Konstantendefinitionen
'
True!=-1
False!=0
'
' ------------------------------------------------------------------
' ------------ Beginn des Hauptprogramms zum ------------------------
' ------------ Testen des Moduls FORMEL.MRG: ------------------------
' ------------------------------------------------------------------
'
Cls
Print
Print
Print "                  F O R M E L B E R E C H N U N G "
Print "                 ___________________________________"
Print
```

2

```
Print "                     (c) 1986 H. Bauch"
Print
First!=True!
@Def_funktion
'
Do
  Input " x = ",X$
  If Val?(X$)<>0
    X=Val(X$)
    @Berechnung(Formel$,X)
    Print " f(x) = ";Berechnung,
    Print "Für neue Funkion <N> eingeben !"
    Print
  Else
    If Upper$(X$)="N"
      @Def_funktion
    Endif
  Endif
Loop
'
' -------------------------------------------------------------------
' ------------ Ende des Hauptprogramms ------------------------------
' -------------------------------------------------------------------
'
Procedure Def_funktion
  '
  ' -----------------------------------------------------------------
  '  Dieses Unterprogramm dient nur zu Testzwecken; es ermöglicht   '
  '  erstmaliges und wiederholtes Definieren einer Formel/Funktion in '
  '  Form eines Strings. Funktionen können auch konstant sein, d. h. '
  '  es kommt keine Variable (hier "x") in ihr vor.                  '
  ' -----------------------------------------------------------------
  '
  ' Globale Variable: True!,False!,First!,F_max%,Std_funktion$(),Formel$
  '                   Fehler_positon%
  Local I%,Xtest,Fehleraus$
  '
  If First!
    @Init_berechnung
    First!=False!
    Print
    Print " Definition einer beliebigen Funktion in BASIC-Syntax"
    Print
    Print " Standardfunktionen: "
    Print
    Print "  / ";
    For I%=1 To F_max%
      Print Std_funktion$(I%);" / ";
    Next I%
  Endif
  Print
  Print
  Repeat
    Input " f(x) = ",Formel$
    ' ---------------------------------------------------------------
    '  An dieser Stelle wird die @Berechnung() zunächst mit einem Test- '
    '  wert aufgerufen; in einigen Fällen kann es bei dem hier gewählten'
    '  Wert zum Fehlerabbruch kommen, weil sich z. B. Division durch   '
    '  Null oder Wurzel aus negativer Zahl ergibt.                     '
    '  Ein einfacher Syntaxfehler wird so jedoch aufgedeckt.           '
    ' ---------------------------------------------------------------
    Xtest=0.0123
    @Berechnung(Formel$,Xtest)
    If (Fehler_position%<>0)
      Fehleraus$="Fehler in der Formel|an Position "+Str$(Fehler_position%
      Alert 1,Fehleraus$,1," Weiter ",Dummy
    Endif
```



3

```
  Until (Fehler_position%=0)
Return
'
' ********************************************************************
' *                                                                  *
' *                     --- FORMEL.MRG ---                           *
' *        Berechnung einer beliebigen Funktion als String           *
' *                                                                  *
' ********************************************************************
'
Procedure Init_berechnung
  '
  ' ------------------------------------------------------------------
  '  Dieses Unterprogramm dient zur Initialisierung der benutzbaren  '
  '  Standardfunktionen für die Procedure Faktor; Erweiterungen können '
  '  dort leicht eingefügt werden, Funktionsnamen müssen in der DATA-  '
  '  Zeile ergänzt werden und F_max% muß angepaßt werden.            '
  '  Init_berechnung muß stets vor der Procedure Berechnung aufgerufen '
  '  werden.
  ' ------------------------------------------------------------------
  '
  ' Globale Variable: F_max%,
  '                   Std_funktion$()
  Local I%
  '
  ' Anzahl z.Zt. implementierter Standardfunktionen:
  F_max%=9
  '
  ' Standardfunktionen festlegen
  Dim Std_funktion$(F_max%)
  Data ABS,SQR,SIN,COS,TAN,ATN,LN,LOG,EXP
  Restore
  For I%=1 To F_max%
    Read Std_funktion$(I%)
  Next I%
Return
'
'
Procedure Berechnung(Funktion$,X)
  '
  ' ------------------------------------------------------------------
  '  Dies ist die Hauptroutine des Moduls FORMEL.MRG; nach Aufruf von  '
  '  von Init_berechnung wird stets hier eingesprungen. In einem Haupt- '
  '  programm können verschiedene Formeln benötigt werden; diese werden '
  '  einfach jeweils neu durch den Stringparameter Funktion$ übergeben. '
  '  Z. Zt. ist nur eine Variable (X) in der Procedure Faktor vorge-  '
  '  sehen, Erweiterungen auf beliebig viele Variable sind möglich    '
  '  durch Ergänzung der Parameterliste und Erweiterung der Procedure '
  '  Faktor an der dort bezeichneten Stelle.                          '
  '                                                                   '
  '  Sollte in Funktion$ ein Syntax-Fehler gefunden werden, so wird   '
  '  Fehler position% als Global zurückgegeben.                       '
  '                                                                   '
  '  Da GfA-BASIC keine Function im Sinne von PASCAL besitzt, werden die'
  '  Ergebnisse einer Procedure als Variable mit dem Namen der Proce- '
  '  dure zurückgegeben.
  ' ------------------------------------------------------------------
  '
  ' Globale Variable: True!, False!, F_max%, Berechnung, Fehler_position%
  '                   Std_funktion$(), Berechnung
  '
  Local Zeil_ende$,Position,Zei$,Ausdruck
  '
  Zeil_ende$=Chr$(0)
  Position%=0
  @N_zeichen
  @Ausdruck
```

4

```
    Berechnung=Ausdruck
    If Zei$=Zeil_ende$
      Fehler_position%=0
    Else
      Fehler_position%=Position%
    Endif
Return
'
'
Procedure Ausdruck
  '
  Local E,Einf_ausdruck,Operator$
  '
  @Einf_ausdruck
  E=Einf_ausdruck
  While Zei$="+" Or Zei$="-"
    Operator$=Zei$
    @N_zeichen
    If Operator$="+"
      @Einf_ausdruck
      E=E+Einf_ausdruck
    Endif
    If Operator$="-"
      @Einf_ausdruck
      E=E-Einf_ausdruck
    Endif
  Wend
  Ausdruck=E
Return
'
'
Procedure Einf_ausdruck
  '
  Local S,Term,Operator$
  '
  @Term
  S=Term
  While Zei$="*" Or Zei$="/"
    Operator$=Zei$
    @N_zeichen
    If Operator$="*"
      @Term
      S=S*Term
    Endif
    If Operator$="/"
      @Term
      S=S/Term
    Endif
  Wend
  Einf_ausdruck=S
Return
'
'
Procedure Term
  '
  Local T,Vorz_faktor
  '
  @Vorz_faktor
  T=Vorz_faktor
  While Zei$="^"
    @N_zeichen
    @Vorz_faktor
    T=T^Vorz_faktor
  Wend
  Term=T
Return
'
```

5

```
Procedure Vorz_faktor
  '
  Local Faktor
  '
  If Zei$="-"
    @N_zeichen
    @Faktor
    Vorz_faktor=-Faktor
  Else
    @Faktor
    Vorz_faktor=Faktor
  Endif
Return
'
'
Procedure Faktor
  '
  Local F,I%,L%,Beginn%
  '
  If Zei$>="0" And Zei$<="9"
    ' ------------------------------------------------------------------
    '  "Zusammenbasteln" einer Zahlen-Konstante                        '
    ' ------------------------------------------------------------------
    Beginn%=Position%
    Repeat
      @N_zeichen
    Until Not (Zei$>="0" And Zei$<="9")
    If Zei$="."
      Repeat
        @N_zeichen
      Until Not (Zei$>="0" And Zei$<="9")
    Endif
    If Upper$(Zei$)="E"
      @N_zeichen
      Repeat
        @N_zeichen
      Until Not (Zei$>="0" And Zei$<="9")
    Endif
    F=Val(Mid$(Funktion$,Beginn%,Position%-Beginn%))
  Else
    If Zei$="("
      ' ----------------------------------------------------------------
      ' Faktor ist ein geklammerter Ausdruck, d.h.                     '
      ' rekursiver Aufruf der Procedure Ausdruck                       '
      ' ----------------------------------------------------------------
      @N_zeichen
      @Ausdruck
      F=Ausdruck
      If Zei$=")"
        @N_zeichen
      Else
        Fehler_position%=Position%
      Endif
    Else
      If Upper$(Zei$)="X"
        ' --------------------------------------------------------------
        ' Variable der Formel gefunden!                                '
        ' Hier können Erweiterungen für beliebig viele Variable        '
        ' "eingebaut" werden; siehe auch Kommentar zu Beginn der       '
        ' Hauptroutine Procedure Berechnung().                         '
        ' --------------------------------------------------------------
        @N_zeichen
        F=X
      Else
        ' --------------------------------------------------------------
        ' Suche nach Standardfunktions-Namen wie bei                   '
        ' Procedure Init_Berechnung in der Data-Zeile vereinbart!      '
        ' Hier können noch beliebig viele Funktionen ergänzt werden    '
```

6

```
        ' (z.B. arcsin, arccos, Fakultät, hyperbol. Funktion usw.).    '
        ' --------------------------------------------------------------
        Erfolg!=False!
        For I%=1 To F_max%
          If Erfolg!=False!
            L%=Len(Std_funktion$(I%))
            If Upper$(Mid$(Funktion$,Position%,L%))=Std_funktion$(I%)
              Position%=Position%+L%-1
              @N_zeichen
              @Faktor
              F=Faktor
              If I%=1
                F=Abs(F)
              Endif
              If I%=2
                F=Sqr(F)
              Endif
              If I%=3
                F=Sin(F)
              Endif
              If I%=4
                F=Cos(F)
              Endif
              If I%=5
                F=Tan(F)
              Endif
              If I%=6
                F=Atn(F)
              Endif
              If I%=7
                F=Log(F)
              Endif
              If I%=8
                F=Log10(F)
              Endif
              If I%=9
                      F=Exp(F)
                    Endif
                    Erfolg!=True!
                  Endif
                Endif
              Next I%
              If Not Erfolg!
                Fehler_position%=Position%
              Endif
            Endif
          Endif
          '
    Endif
    Faktor=F
Return
'
'
Procedure N_zeichen
  '
  Repeat
    Inc Position%
    If Position%<=Len(Funktion$)
      Zei$=Mid$(Funktion$,Position%,1)
    Else
      Zei$=Zeil_ende$
    Endif
  Until Zei$<>" "
Return
'
'
' ------------------------------------------------------------------------
' ----------- Ende des Moduls FORMEL.MRG ---------------------------------
' ------------------------------------------------------------------------
```

Assemblerkurs

Teil 3

Dieser Teil des Assemblerkurses soll Ihnen ein paar Gruppen des Befehlssatzes des 68000 mit einigen Anwendungen zeigen.

In unserem letzten Programm (Ausgabe 1/86), haben Sie ein Beispiel für die Anwendung der Arithmetik- und Programmsteuerbefehle gesehen. Außerdem konnte man die Umsetzung eines Algorithmus am Beispiel des Hornerschemas sehen.

Als Übersicht, gebe ich Ihnen an dieser Stelle, die Bezeichnungen der Gruppen, die in diesem Teil des Kurses besprochen werden.

- Arithmetische Befehle
- BCD Befehle
- Logische Befehle

Die arithmetischen Befehle

Die Gruppe der arithmetischen Befehle umfaßt außer den vier Grundrechenarten, die Negation und die vorzeichenrichtige Erweiterung, die virtuelle Substraktion (Vergleich). Hier nun die Tabelle der Befehle.

ADD	binäre Addition
CLR	löschen eines Operanden
CMP	vergleichen zweier Operanden
DIV	Division
EXT	vorzeichenrichtige Erweiterung
MUL	Multiplikation
NEG	Zweierkomplement
SUB	binäre Subtratkion
TST	testen gegen Null

Die Addition ADD

Dieser Befehl existiert in vier Versionen. Die Unterscheidung wird durch das Anhängen eines Buchstabens kenntlich gemacht. Das hatten Sie ja schon bei dem MOVE Befehl kennengelernt. Bei dem ADD-Befehl gibt es aber noch die Variante ADDX. Er hat die Funktion wie der ADC-Befehl beim 6502 Prozessor. Das X-Flag (Extended) wird bei bestimmten Befehlen genauso gesetzt, wie das C-Flag (Carry). Andere Befehle beeinflussen jedoch nur das C-Flag, so daß es möglich ist, daß X-Flag über einige Operationen aufzuheben.

Formen des ADD Befehls

ADD	normale Addition
ADDA	auf Adressregister
ADDI	mit Konstante
ADDQ	addiere schnell
ADDX	addiere mit X-Flag

Beispiel:

```
ZAHL1 EQU 17
ZAHL2 EQU $0D

        MOVEI.B   #ZAHL1,D1
        MOVEI.B   #ZAHL2,D2
        ADD.B     D2,D1

        MOVEI.B   #ZAHL1,D1
        ADDI.B    #Zahl2,D1

        MOVEA.L   (SP)+,A3
        ADDA.L    (SP)+,A3

        ADD.L     D1,D3
        ADDX.L    D2,D4
```

In diesen Beispielen sieht man den Umgang mit dem ADD-Befehl. Als erstes, wird den „Variablen" ZAHL1 und ZAHL2 mit der EQU-Anweisung (Equal = Gleich) die Werte 17 und $0D zugewiesen. Manche Assembler erlauben auch ein „=" Zeichen an dieser Stelle. Da der ADD-Befehl immer ein Datenregister als Quelle oder Ziel haben muß, speichern wir die Werte in Datenregistern. Die Operation, die bei dieser Art von Befehlen ausgeführt wird, ist immer QUELLE Operation ZIEL nach ZIEL. Somit erscheint als Ergebnis im Datenregister D1 der Wert 30 bzw. $1E. Die Flags werden entsprechend dem Ergebnis gesetzt (Alle auf NULL).

Mit dem nächsten Beispiel sparen wir uns einen Befehl und ein Datenregister. Am Ergebnis ändert sich jedoch nichts.

Als nächstes werden zwei Adressen vom Stack miteinander addiert, deren Summe sich im Adressregister A3 befindet. Die Flags werden von dieser Operation nicht berührt. Dies bedeutet, daß ein eventuell entstehender Überlauf nicht erkannt wird.

Das letzte Beispiel zeigt eine 64 Bit Addition, wobei die erste Zahl mit ihrer Bitfolge in D2,D1 steht, und die zweite Zahl in D4,D3. Als erstes werden die beiden niederwertigen Langworte addiert, und danach die zwei höherwertigen Langworte, wobei der ADDX-Befehl einen eventuellen Überlauf mit berücksichtigt. Das Ergebnis steht dann in den Registern D4,D3. Ist bei der zweiten Addition ein Überlauf aufgetreten, so ist dies wieder durch das X-Flag vermerkt.

Die Subtraktion SUB

Ganz analog zu der Addition verläuft die Subtraktion, wenn man beachtet, daß das X-Flag bei der Subtraktion eine Entlehnung (Borgen) bedeutet. Mit em ADDX und SUBX werden folgende Operationen ausgeführt:

ADD
Quelle + Ziel + X Flag = Ziel
SUB
Quelle − Ziel − X Flag = Ziel

Beim SUBX-Befehl hingegen, wird das X− Flag mit abgezogen, wenn es gesetzt ist.

Der Vergleich CMP

Will man etwas über den Zustand eines Registers oder einer Speicherstelle bezüglich einer weiteren Größe erfahren, so bietet sich der CMP-Befehl (compare = vergleichen) an. Dieser Befehl bewirkt eine virtuelle Subtraktion. Das bedeutet, er subtrahiert die eine von der anderen Größe und setzt entsprechend dem Ergebnis die Flags. Das Ergebnis wird allerdings nicht, wie beim SUB-Befehl, an das angegebene Ziel gespeichert.

Auch hier werden weitere Buchstaben angehängt, die die Anwendung deutlicher machen sollen. Bei den meisten

Assemblern ist die Schreibweise redundant, das heißt, der Assembler erkennt sowohl

CMP (A1)+,(A2)+

als auch

CMPM.W (A1)+,(A2)+

als equivalente Schreibweise an. Ebenso wie der Assembler als Standardverarbeitung Wortverarbeitung wählt. Zugelassen als Schreibweisen sind:

CMP	normaler Vergleich
CMPA	auf Adressregister
CMPI	mit Konstante
CMPM	im Speicher

Beispiele:

```
     CMPI.B   #65,D3
     BEQ      GEFUNDEN

     MOVEA.L  #BLOCK1,A1
     MOVEA.L  #BLOCK2,A2
     MOVEI.L  #ANZAHL-1,D1
LOOP CMPM.B   (A1)+,(A2)+
     BNE      UNGLEICH
     DBRA     D1,LOOP
```

Da der CMP-Befehl alle Flags – außer dem X-Flag – beeinflußt, wird er fast ausschließlich mit einer bedingten Verzweigung benutzt. Das erste Beispiel verzweigt dann, wenn im ersten Byte von D3 dezimal 65 steht. Im zweiten Beispiel werden erst einmal die Anfangsadressen von zwei Blöcken festgelegt. Danach benötigt man noch die Länge der Blöcke minus eins, wegen der Funktion des DBcc-Befehls. Dann werden die Blöcke solange miteinander verglichen, bis die Anzahl (D1) gleich –1 ist, oder bis ein Fehler (Ungleichheit) festgestellt worden ist. Die Adressen der „Ungleichheit" ist dann um eins kleiner als A1 und A2.

Löschen CLR

Dies ist ein recht einfacher Befehl, da er nur den Zieloperaden besitzt. Den Vorteil, den der CLR-Befehl hat, ist seine Geschwindigkeit gegenüber vergleichbaren Konstruktionen. Mit ihm kann man recht einfach so ziemlich alles löschen (Null setzen).

Beispiel:

```
CLR.L  D0
CLR.W  $85000
CLR.L  (A3)
```

Erweitern EXT

Dieser Befehl wird benutzt um Datenregister vorzeichenrichtig zu erweitern. Dies ist manchmal nötig, wenn z. B. ein Datenregister nur ein Byte oder Wort als Adresse beinhaltet, aber als Indexregister Wort- oder Langwortlänge haben muß.

Als Verarbeitung ist nur Wort und Langwort zugelassen. Ist Wortverarbeitung gewählt, so wird ein Byte vorzeichenrichtig auf Wort erweitert. Ebenso verhält es sich mit Wort auf Langwort. Damit kann man auch durch zweimaliges Anwenden von EXT auf das Datenregister, von Byte auf Langwort erweitern. Erweitert wird, indem das höchstwertigste Bit (Bit 7 oder 15) in die restlichen Stellen (Bit 8–15 oder Bit 16–31) kopiert wird.

Beispiel:

EXT.L D0

	vorher	nachher
D0	XXXX7FD3	00007FD3
D0	XXXX8FD3	11118FD3

EXT.W D4

	vorher	nachher
D4	12435687	1243FF87
D4	12345678	12340078

Anhand dieser Beispiele können Sie noch einmal die Wirkungsweise von Wort- und Langwortverarbeitung sehen. Die Grenze, an der sich entscheidet, ob mit Eins oder Null erweitert wird, ist die Sieben. Zahlen, die an der entsprechenden Stelle größer sieben sind, werden mit eins erweitert.

Die Negation NEG

Die Negation (Zweierkomplement) dient zum Vorzeichenwechsel von Binärzahlen. Dies erreicht der Befehl durch eine Subtraktion, indem er die Zahl von Null abzieht. Die Flags werden bei dieser Operation entsprechend dem Ergebnis gesetzt.

Um Zahlen, die größer als 32 Bit lang sind, benötigt man den NEGX-Befehl, da dann die Entlehnung der Subtraktion mit berücksichtigt wird. In dem Beispiel sieht man die Wirkung des X-Flags auf die Negation. Das X-Flag wird einfach ebenfalls von Null abgezogen.

Beispiel:

NEG.W D0

	vorher	nachher
D0	XXXX4321	XXXXBCDF

NEGX.W (A3)

	vorher	nachher
A3	00006000	00006000
X=0		
$6000	1AC2	E53E
X=1		
$6000	1AC2	E53D

Die Multiplikation MUL

Eine der Stärken des 68000 ist die Multiplikation und Division. Diese beiden Befehle erlauben die Verarbeitung von vorzeichenbehafteten (signed) und vorzeichenlosen (unsigned) Zahlen. Die Schreibweisen dafür sind MULS und MULU. Die Operandengröße ist auf 16 Bit festgelegt. Als Ergebnis erscheint eine 32 Bit Zahl. Die Flags zeigen nach der Operation an, ob das Ergebnis negativ oder null ist.

Beispiel:

MULI #10,D3

	vorher	nachher
D3	XXXX2525	00074040

MULS (A3),D2

	vorher	nachher
A3	00001450	00001450
D2	XXXX2837	FFFFE7D9
$1450	FFFF	FFFF

Im ersten Teil erfolgt eine vorzeichenlose Multiplikation mit einer Konstanten. Da diese im dezimalen Zahlensystem angegeben ist, findet man diese auf den ersten Blick im Ergebnis nicht wieder. Das nächste Beispiel ist eine vorzeichenbehaftete Multiplikation mit der Adressierung Adressregister indirekt (ARI). Das Adressregister zeigt auf die Adresse des Multiplikanten. Der Multiplikant hat den Wert –1, da er vorzeichenbehaftet ist. Dies entspricht einer Negation.

Die Division DIV

Auch hier existieren die zwei Versionen vorzeichenlos und -behaftet. Die Division entspricht dem MOD-Befehl in Pascal. Es ist eine ganzzahlige Divi-

sion mit Rest. Die Operation, die der Befehl ausführt, ist eine Division einer 32 Bit Zahl durch eine 16 Bit Zahl. Das Ergebnis im Datenregister besteht aus 16 Bit (0–15) Divisor und dem Rest (16–31). Bei der Division können zwei außergewöhnliche Fälle auftreten. Erstens, es wird versucht durch Null zu teilen, dann wird sofort in eine Exception verzweigt. Zweitens, der Quotient würde größer als 16 Bit werden, dann wird die Operation nicht ausgeführt. Der Divisor und der Dividend bleiben unverändert. Die fehlerhafte Operation wird im V-Flag (Overflow) kenntlich gemacht.

Beispiel:

DIVU #3,D3

	vorher	nachher
D3	0000000E	00020004

Anhand des Beispiels kann man die ganzzahlige Division erkennen. In D3 steht $0E. Dies entspricht Dezimal 14. Teilt man die Zahl durch 3, so ergibt dies vier Rest zwei.

Als letzten Befehl dieser Gruppe haben wir den TST-Befehl.

Testen TST

Dieser Befehl dient zum Prüfen eines Operanden. Der Zieloperand wird mit null verglichen, und danach werden die Flags entsprechend gesetzt. Durch diesen Test kann man eine Aussage bezüglich null machen. Die Wirkungsweise entspricht dem CMPI #0, wobei aber der TST-Befehl schneller ist.

Nun haben wir die Gruppe der arithmetischen Befehle geschafft. Hier ist jetzt die Tabelle mit den Adressierungsarten und den Flags.

In der letzten Ausgabe ist eine Kleinigkeit verloren gegangen. Diese Kleinigkeit ist der Umgang mit der Tabelle. In der Tabelle befindet sich der Syntax nach dem Motorola Standard. Diese Schreibweise ist eine Art „Norm", der von den meisten Assemblern verstanden wird. In der Beschreibung der Adressierungsarten bedeutet der Schrägstrich / „ohne, außer". Z. B. bedeutet Alle/An/PCR, daß alle Adressierungsarten, außer Adressregister direkt und Programmzähler relativ zugelassen sind. Ein Komma hingegen bedeutet „und".

Syntax		Flags XNZVC	.x	Quelle	Ziel
ADD.x	(ea),Dn	*****	B,W,L	Alle	Dn
ADD.x	Dn,(ea)	*****	B,W,L	Dn	Alle/An/PCR
ADDA.x	(ea),An	-----	W,L	Alle	An
ADDI.x	#Konst,(ea)	*****	B,W,L	#	Alle/An/PCR
ADDQ.x	#Konst,(ea)	*****	B,W,L	#	Alle/PCR
ADDX.x	Dy,Dx	*****	B,W,L	Dn	Dn
ADDX.x	-(Ax),-(Ay)	*****	B,W,L	-(An)	-(An)
CLR.x	(ea)	X0000	B,W,L		Alle/An/PCR
CMP.x	(ea),Dn	X****	B,W,L	Alle	Dn
CMPA.x	(ea),An	X****	W,L	Alle	An
CMPI.x	#Konst,(ea)	X****	B,W,L	#	Alle/An/PCR
CMPM.x	(Ay)+,(Ax)+	X****	B,W,L	(An)+	(An)+
DIV	(ea),Dn	X***0		Alle/An	Dn
EXT.x	Dn	X**00	W,L		Dn
MUL	(ea),Dn	X***0		Alle/Dn	Dn
NEG.x	(ea)	*****	B,W,L		Alle/An/PCR
NEGX.x	(ea)	*****	B,W,L		Alle/An/PCR
TST.x	(ea)	***00	B,W,L		Alle/An/PCR

- – bleibt unverändert
- ★ wird entsprechend gesetzt
- U Undefiniert
- 0 wird auf Null gesetzt

Bei der Schreibweise der erlaubten Adressierungsarten bedeutet:

ARI	alle Adressregister indirekt
abs	Absolut kurz und lang
PCR	alle Programmcounter relativ
SR	Statusregister
CCR	Condition Code Register
USP	User Stack Pointer

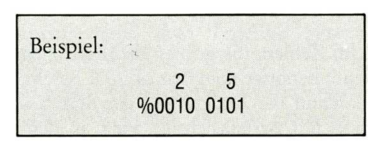
Um alle Adressierungsarten zu ermitteln, kann man jede Quelle mit jedem Ziel verknüpfen.

Der SUB-Befehl hat die gleichen Adressierungsarten wie der ADD-Befehl.

Die nächste Gruppe könnte man auch zu den arithmetischen Befehlen zählen. Da mit ihnen aber keine Binärzahlen bearbeitet werden, hat man sie in eine eigene Gruppe zusammengefaßt.

Die BCD Befehle

BCD bedeutet Binary Coded Dezimal. Dabei wird jede Ziffer Dual, mit der Wertigkeit 8–4–2–1, kodiert. Diese Darstellung der Zahlen findet seine Anwendung hauptsächlich in der Speicherung und Verarbeitung von Gleitkommazahlen.


Beispiel:

2 5
%0010 0101

Der Aufbau von Gleitkommazahlen

Es gibt eine ganze Reihe Darstellungsmöglichkeiten der Gleitkommazahlen. Ich will Ihnen hier aber nur eine vorstellen. Eine Gleitkommazahl besteht aus einem Exponenten und einer Mantisse, (z. B. 1.5E3 = 1500), wobei E (Exponent) gleichbedeutend mit „zehn hoch" ist. Außerdem werden noch zwei Vorzeichen benötigt, und zwar für Exponent und Mantisse. Als einfachste Lösung könnte man den Exponent als Zahl im Zweierkomplement betrachten, und ein Extrabit für das Vorzeichen der Mantisse reservieren.

Da die Rechengenauigkeit von der Mantisse abhängt, speichert man diese ab der ersten von Null verschiedenen Stelle ab. Den Exponenten muß man dann entsprechend der Anzahl, um die man das Komma verschoben hat, angleichen.

15.345E4 = 1.5345E5

oder

15.345E4 = 0.15345E6

Diese Darstellung nennt man die Normalform, wobei entweder die eine oder andere Darstellung verwendet wird. Benutzt man die zweite Normalform, so wird man die Null nicht mit abspeichern, sondern wie gesagt, die erste von Null verschiedene. So sieht nun das Ergebnis der Zahl 1.5E3 im Speicher aus, wenn der Exponent 7-Bit, das Vorzeichen 1-Bit und die Mantisse 4-Byte (8 Stellen Rechengenauigkeit) hat:

%0000011 1 0001 0101 0000 0000
Exponent VZ Mantisse

Nun wieder zurück zu den BCD-Befehlen. Diese Gruppe besteht aus den drei Befehlen:

ABCD Addition

Profitieren Sie von unserem Know-How!

OMIKRON-BASIC



ARITHMETIK
- Rechengenauigkeit bis 19 Stellen bei allen Funktionen
- Rechenbereich bis 5.11 E±4931

GEM
- Komplette GEM-Library
- sämtliche AES- & VDI-Funktionen direkt mit Namen verfügbar
- eigene BITBLIT-Routine
- NEU: jetzt mit Resource Construction Set

EXTRAS
- Masken-INPUT
- SORT-Befehl sortiert beliebige Felder
- Matrizenbefehle
- Compiler ab Frühjahr verfügbar

STRUKTUR
- Prozeduren und mehrzeilige Funktionen
- mit Übergabe- und Rückgabe-Parametern und lokalen Variablen
- REPEAT...UNTIL, WHILE...WEND, mehrzeiliges IF...THEN...ELSE...ENDIF
- Labels bei GOTO, GOSUB, ON...GOTO etc.

KOMPATIBILITÄT
- 99 % MBASIC-kompatibel
- Editor findet Inkompatibilitäten
- dadurch einfachste Anpassung

Editor
- Mit oder ohne Zeilennummern (umschaltbar)
- Drei Schriftgrößen bis 57x128
- Frei definierbare Funktionstasten

GESCHWINDIGKEIT
- FIT-Code (FIT = Fast Interpreting Technique)
- Wir kennen keine schnelleren 68000-Fließkommaroutinen
- Volle Integer-Arithmetik
- eigene Disk-Routinen für beschleunigten Dateizugriff

LIEFERUMFANG
- Modul mit OMIKRON-BASIC (wird seitlich eingesteckt)
- Demodiskette mit Runtime-Interpreter, Hilfsprogrammen und vielen Beispielprogrammen
- 180-seitiges, spiralgebundenes deutsches Handbuch
- Alles zus. nur DM 229,– (unverbindl. Preisempfehlung) Handbuch vorab DM 30,–

„Atemberaubende Geschwindigkeit" (DATA WELT 12/86, S. 69)
„Für ernsthafte Programmierer, die effektive und vor allem schnelle Programme erstellen wollen, ..., ist dieser neue BASIC-Interpreter genau das richtige" (DATA WELT 12/86, S. 70)
„OMIKRON-BASIC kann wesentlich mehr als der BASIC-Standard" (ST Computer 12/86, S. 79)

OMIKRON **Software · Erlachstr. 15 · 7534 Birkenfeld · Tel. (0 70 82) 53 86**

Österreich: Ueberreuter Media, Alser Str. 24, A-1091 Wien
Schweiz: Microtron, Postfach 40, CH-4542 Pieterlen

NBCD Negation
SBCD Subtraktion

BCD-Addition ABCD

Bei dem Umgang mit diesen Befehlen sollte man folgendes berücksichtigen. Die Verarbeitung erfolgt nur mit Byteverarbeitung, also zwei BCD-Zahlen (= gepackte BCD-Zahlen). Es wird stets das X-Flag mitverarbeitet, und es sollte darauf geachtet werden, daß nur BCD-Zahlen mit dieser Gruppe verarbeitet werden. Eine weitere Besonderheit ist das Z-Bit (Zero), da es nur gelöscht wird, wenn das Ergebnis ungleich null ist. Dies bedeutet, daß man das Z-Flag vor der Operation auf eins setzen muß! Eine BCD-Zahl besteht aus vier Bits, wobei aber nur die Zahlen von 0 bis 9 Dual kodiert werden. Die restlichen Kombinationen werden nicht benötigt. Man nennt sie Pseudotetraden.

Beispiel

```
        MOVE.B #%01000101,D1   ;45
        MOVE.B #%00110110,D2   ;36
        ABCD D2,D1

D1 = %10000001 (81)
```

Die Addition erfolgt zuerst mit den niederwertigsten vier Bits. Dies ergibt 5+6 gleich 11. Da aber bei 10 schon ein Übertrag stattfindet, bleibt eins übrig. Bei der Addition der nächsten BCD-Zahl wird dieser Zwischenübertrag berücksichtigt. Somit ergibt die nächste Teilrechnung: 4+3+1 gleich 8. Da sich hier kein Übertrag ergab, wird das C- und X-Flag gelöscht. Das Z-Flag wird nur dann gelöscht, wenn es gesetzt war.

BCD-Subtraktion SBCD

Ebenso wie beim ABCD-Befehl gelten hier die Besonderheiten in Bezug auf die Verarbeitung und der Flags. Nur bedeutet ein gesetztes X-Flag kein Übertrag, sondern ein Borgen. Das hatten wir ja schon beim SUB-Befehl.

Hier ist ein Beispiel für die Verarbeitung von mehr als zwei BCD-Zahlen. Da dies einfacher im Speicher als in den Registern zu lösen ist, nehmen wir einmal an, sie stünden an den entsprechenden Adressen. Als erstes werden zwei Adressregister mit der Adresse des Zahlenendes plus eins geladen, danach löschen wir das X-Flag und setzen das Z-Bit im CCR. Dadurch wird ein vorhandener Übertrag nicht mit subtrahiert und die Null als Ergebnis abfragbar. Nach der Operation befindet sich an der Speicherstelle $1000 das Ergebnis. Außerdem ist das X-Flag gesetzt worden, was bedeutet, daß ein Borgen stattgefunden hat. Dies wiederum bedeutet, das die Zahl negativ ist. Nun gibt es aber keine negativen BCD-Zahlen. Das endgültige Ergebnis wäre somit um diesen Betrag kleiner als die größte mit sechs Ziffern nicht mehr darstellbare Zahl. Subtrahieren sie von dem Ergebnis einfach 1 000 000 mit dem Taschenrechner, und sie erhalten das gleiche Ergebnis, als wenn sie es nur mit dem Taschenrechner gerechnet hätten.

Beispiel:

```
Zahl1:123456
Zahl2:835933

Im Speicher:

1000 12
1001 34
1002 56

2000 83
2001 59
2002 33

                MOVEA #$1003,A1
                MOVEA #$2003,A2

                MOVE  #4,CCR

                SBCD  -(A2),-(A1)
                SBCD  -(A2),-(A1)
                SBCD  -(A2),-(A1)

Nach der Operation

1000 28
1001 75
1002 23
```

BCD-Negation NBCD

Mit diesem Befehl können Sie das Neuner- bzw. Zehnerkomplement bilden. Ist das X-Flag vor der Operation gesetzt, so wird das Neunerkomplement erzeugt. Auch hier gelten wieder die Besonderheiten der BCD-Befehle. Mit diesem Befehl kann man negative Zahlen in BCD-Darstellung erzeugen. Wendet man dies auf das Ergebnis des SBCB-Beispiels an, so nimmt das Zehnerkomplement, da die Zahl negiert werden soll.

Beispiel:

```
                MOVEA #$1003,A1
                MOVE  #4,CCR

                NBCD  -(A1)
                NBCD  -(A1)
                NBCD  -(A1)

1000 71
1001 24
1002 77
```

Da an der Zahl nicht zu erkennen ist, ob es sich um eine negative Zahl handelt, sollte man dies noch irgendwo vermerken.

Die logischen Befehle

Mit dieser Gruppe von Befehlen können Sie logische Verknüpfungen bewerkstelligen; sei es, daß Sie diese nur zum Setzen oder Ausblenden von Bits benutzen oder um Boolsche Ergebnisse berechnen zu wollen. Zu den Boolschen Operationen gehören:

- AND logisches UND
- EOR Exclusiv ODER
- NOT invertieren
- OR logisches ODER

Das Und (AND)

Mit dem AND-Befehl kann man zwei Operanden miteinander „verunden". Dieses Verunden findet bitweise statt. Die Flags werden nach der Operation dem Ergebnis entsprechend gesetzt oder gelöscht. Das bitweise verunden stellt man mit einer Wahrheitstabelle dar. Eines der wichtigsten Befehle ist das verunden einer Konstanten mit dem CCR, bzw. dem SR. Bei der BCD-Subtraktion hatten wir den Befehl MOVE #4,CCR. Dadurch wurden alle Flags gelöscht und das Z-Flag gesetzt. Will man aber nur ein oder mehrere Flags gezielt löschen und den

Syntax	Flags XNZVC	.x	Quelle	Ziel
ABCD Dy,Dx	*U*U*		Dn	Dn
ABCD -(Ay),-(Ax)	*U*U*		-(An)	-(An)
SBCD Dy,Dx	*U*U*		Dn	Dn
SBCD -(Ay),-(Ax)	*U*U*		-(An)	-(An)
NBCD (ea)	*U*U*			Alle/An/PCR

Rest unbeeinflußt lassen, so kann man dies mit der AND-Verknüpfung machen.

Wahrheitstabelle

B	A	Q
0	0	0
0	1	0
1	0	0
1	1	1

Formen von AND

AND
Zwei Operanden

ANDI
Konstante mit Operand

ANDI to CCR
Konstante mit CCR

ANDI to SR
Konstante mit SR (PRIV)

Beispiel:

```
D1        %01100101
ANDI.B    #%11001001,D1
D1 =      %01000001
```

Immer nur dann, wenn zwei Einsen verundet werden, erscheint eine Eins im Ergebnis. Mit dem AND-Befehl können Sie bestimmte Bits ausblenden (löschen) oder andere Bits abfragen. Alle Operationen, die das oberste Byte des SR verändern können, sind Privilegierte Befehle (Priv), die im Usermode nicht benutzt werden dürfen.

Das Oder (OR)

Ebenso wie mit dem AND-Befehl funktioniert dies mit allen anderen logischen Operationen. Das OR bildet das Gegenstück zum AND. Mit ihm werden bestimmte Bits gesetzt, wobei die restlichen Bits erhalten bleiben. Der OR-Befehl hat die gleichen Formen, wie der AND-Befehl.

Wahrheitstabelle

B	A	Q
0	0	0
0	1	1
1	0	1
1	1	1

Beispiel:

```
D1 =      %01100101
ORI.B     #%11001001,D1
D1 =      %11101101
```

Syntax	Flags XNZVC	.x	Quelle	Ziel
AND.x (ea),Dn	-**00	B,W,L	Alle/An	Dn
AND.x Dn,(ea)	-**00	B,W,L	Dn	Alle/An/PCR
ANDI.x #Kons,(ea)	-**00	B,W,L	#	Alle/An/PCR
NOT.x (ea)	-**00	B,W,L		Alle/An/PCR

Das Exclusiv Oder (EOR)

Das Exclusiv Oder wird auch Antivalenz genannt, weil im Ergebnis dann eine Eins erscheint, wenn die zu verknüpfenden Bits ungleich sind. Man kann es auch so interpretieren, daß die Aussage dann wahr (1) ist, wenn nur die eine oder die andere Bedingung wahr ist. Dieser Befehl wird in Zeichenprogrammen benutzt. Dadurch, daß man zwei Speicherbereiche mit EOR verknüpft, werden zwei überlappende Objekte auf dem Bildschirm sichtbar. Auch hier existieren die gleichen Formen wie vorausgegangen.

Wahrheitstabelle

B	A	Q
0	0	0
0	1	1
1	0	1
1	1	0

Beispiel:

```
D1 =      %01100101
EORI.B    #%11001001,D1
D1 =      %10101100
```

Das Nicht (NOT)

Der NOT-Befehl bildet einfach das Einerkomplement. Dies macht er, indem er aus einer Eins eine Null und umgekehrt macht.

Wahrheitstabelle

A	Q
0	1
1	0

Beispiel:

```
D1 =   %01100101
NOT.B  D1
D1 =   %10011010
```

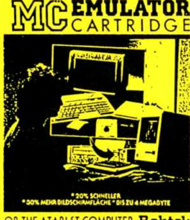
Die Tabelle für AND gilt ebenfalls für OR und EOR.

Für diese Ausgabe ist es wohl erst einmal genug. In der nächsten Ausgabe wird wieder ein Programm für Sie dabei sein. Ich hoffe Sie haben jetzt schon einiges gelernt und sich daraus etwas erarbeiten können.

(Sven Schuler)

GFA-BASIC
Ein Basic, auf das Sie schon lange gewartet haben. Wir erleben täglich helle Begeisterung.
- kompakter Hochgeschwindigkeits-Interpreter.
- 11-stellige Genauigkeit **169,-**
- Strukturiertes Programmieren.
- Einfachste GEM-Programmierung.

GFA-Compiler: so gut wie Basic **169,-**
GFA-Vektor: schnelle 3D Graphik. **149,-**
GFA-Draft: GEM-gesteuertes CAD-Prg Maßstabsgerechte Zeichnungen. Für Plotter und Drucker **298,-**



Der Mac Emulator
Die Chance für Sie, auf Ihrem Atari ST Macintosh Software zu fahren.
- läuft dadurch 20 % schneller als auf dem Mac
- nutzt Ihren Monochrombildschirm voll aus (größer als Macscreen!)
- größere Speicherkapazität als Mac (bei 520 ST+, oder 1040 ST)

Steigen Sie mit dem Mac Emulator in die Welt des Macintosh ein!
DM 495,-

Megamax C
Ein komplettes C-Entwicklungssystem
- Single Pass-Compiler
- Inline Assembler
- Disassembler
- Linker & Librarian
- Resource Construktion
- Vollständige GEM-Libraries
- Unix-Routinen
- GEM-Editor, GEM-Shell
- 370 Seiten-Handbuch

DM 495,-
Händleranfragen erwünscht

Adress-access 2
Die phantastische Adressverwaltung für den Atari ST
- GEM-Funktion
- Menü-Wahl mit Maus oder Funktionstasten
- Gleichzeitig Suchen nach allen Textfeldern
- Sortieren nach beliebigem Feld
- Ausgabe auf Etiketten und Formularen
- Bildschirmausgaben mit vertikalen und horizontalem Scrolling
- Mit deutschem Handbuch für alle Atari ST's **99,-**

Wordstar/Mailmerge
Der Bestseller unter den Textverarbeitungsprogrammen bietet Ihnen bildschirmorientierte Formatierung, deutscher Zeichensatz und DIN-Tastatur sowie integrierte Hilfstexte
- Mailmerge für Serienbriefe
- Beliebige Anzahl Adressen
- Drucken von Adressaufklebern
- Für beliebige Drucker (centronics)

Programm: **199,-**
Buch zum Programm: **49,-**

1st Word Plus **199,-**
Erhebliche Leistungssteigerung in der Textbe- und -verarbeitung und erlaubt Grafikeinbindung in den Text.
- Briefe mit Bildern im Text
- Dokumentation mit Schaubildern oder dig. Fotos im Text
- Sourcecode-Erstell. für Programmiersprachen
- Volle GEM-Unterstützung
- Integration von Text und Grafik
- Rechtschreibprüfung (engl.)
- Textformattrennung / Dezimaltab.
- Komfortabler Suchen u. Ersetzen
- 1st Mail **99,-**

VIP-Professional
(Lotus 1, 2, 3) **DM 648,-**



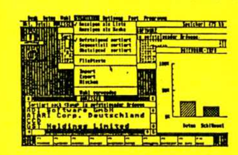

Omikron-Basic
- Extrem schnell - Lieferb. auf Modul
- Prozeduren mit Übergaben und Rückgabenparam. u. lok. Variablen
- 6 Variablentypen
- Im mathemat. Bereich unschlagbar: 19 stellige Rechengenauigkeit 54 mathemat. Funktionen u. sehr schnelle Opertoren
- Unterstützt prof. kaufm. Programmierung, z. B. Masken-Input, Sortierbefehle, Isam-Dateiverwaltung
- VDI- u. AES-Funkt. direkt über GEM-Library aufrufbar
- Echter Direktmod. mit voll. Screen-Editius **229,-**

Boffin
- Neueste Textverarb. unter GEM (15 Minuten Lernzeit)
- Mischt Grafik + Text
- Versch. Schriftarten u. -größen möglich
- Zeichnet Geschäftsgrafiken (Balken-, Liniendiagramme etc.)
- Beherrscht die Grafik

Farbige Grafiken und Zeichnungen aus Degas, Doodle, Neochrom und Metafiles werden eingelesen und fürs Zeichnen während des Textes kann auch das angebotene Zeichenprogr. Kreis, Linie, etc. über ein Pull-Down-Menü aufgerufen werden. **398,-**

Mono-Star
- Pixelorientiertes Programm, also kreativ
- Funktion nur im schw./w.-Modus
- kpl. in GFA-Basic geschrieben
- Wenig benutzte Funktionen sind herausgelassen oder durch neue ersetzt

Color-Star
- wie "Mono Star", aber Farbe
- Farbanimation mit 512 Farben
- GEM unterstützt
- verformen von Zeichnungen
- mischen von Farben

Adimens ST


Echt relationale Datenbank mit GEM-Oberfläche **499,-**
Signum: Textverarbeitung in einer anderen Dimension **448,-**

Cumana
Laufwerke
3,5 Zoll Doppellaufwerk 80 Spuren doppelseitig 1 MB, Anschlußkabel und Netzkabel 699,-

3,5 Zoll Doppellaufwerk 2 x 80 Spuren doppelseitig 2 x 1 MB, Anschlußkabel, und Netzkabel (nicht für Atari 1040 ST) (auch übereinander) **1099,-**

5 1/4 Zoll Slimline-Laufwerk 40/80 Spuren umschaltbar doppelseitig 1 MB komplett mit Netzkabel und Anschlußkabel **759,-**

5 1/4 Zoll Slimline Doppellaufwerk 2 x 40/80 Spuren umschaltbar 2 x 1 MB komplett mit Netzkabel und Anschlußkabel (nicht für Atari 1040 ST) **1299,-**

»mixed« slimline Doppelstation mit einem 3,5 Zoll und einem 5 1/4 Zoll Laufwerk 40/80 Spuren umschaltbar 2 x 1 MB **1199,-**

Atari-kompatible Maus **120,-**

Cumana Laufwerke sind fast unhorbar, lang und kritisch getestet, einfach absolute Spitzenklasse!

Spiele:
1.	Borrowed Time	89,90
2.	Brataccas	89,90
3.	Colourspace	98,-
4.	Delta Patrol	129,-
5.	Dragonworld	149,-
6.	Flightsimulator SUB	178,-
7.	Hichthikers	99,-
8.	Land of Havoe	79,90
9.	Little Computer People	89,90
10.	Major Motion	79,90
11.	Quiwi	69,-
12.	Sun Dog	98,-
13.	Time Bandit	99,-
14.	The Hacker	89,-
15.	The Pawn	99,-
16.	The Black Couldron	99,-
17.	The Music Studio	99,-
18.	Top Secret	89,-
19.	Wintergames	98,-
	außerdem:	
20.	Arena	99,-
21.	Ballyhoo	89,-
22.	Deep Space	139,-
23.	Leader Board	89,-
24.	Starcross	99,-
25.	Silent Service	99,-
26.	The Witness	89,-
27.	Hanse	79,-

Kuma-Programme
K-Seka (Assembler)	**148,-**
K-Spread (Kalkulation, deutsch)	**148,-**
K-Graph (Grafik, deutsch)	**118,-**
K-Comm (deutsch)	**148,-**
K-Resource (Utilities)	**118,-**
K-Word (Text)	**118,-**
K-Ram (Ram-Disk)	**89,-**
K-Switch (Utilities)	**89,-**
K-Minstrel (Musik)	**99,-**
K-Graph II	**148,-**

Achtung! Wir liefern auch gern in die Schweiz und nach Österreich!
Zahlung per Scheck oder Überweisung (Vorkasse) wegen komplizierter Nachnahmeabwicklung erwünscht.

Potsdamer Ring 10 · 7150 Backnang · ☎ 07191/1528-29

weeske
COMPUTER-ELEKTRONIK

G-Data Software
Ausgesuchte Programme mit hohem Gebrauchswert.

G-Harddisk-Help: Sicherheitsduplikat auf Disketten **79,-**
G-Utilities: G-Format, Time, Free, List, Trans, Verify Datamax Prim, RCS Superbackup **65,-**
G-Diskmon: Alles um den Sektor String und Bytes **89,-**

G-Ramdisk: Wird als Desk Accessory bei jedem Systemstart mitgebootet. 100mal schneller als Laufwerke. Nach Reset und neuem boot bleibt Inhalt erhalten. **48,-**

as-adress: Adressverwaltung in komfortabler GEM-Umgebung. 1st-word kompatible Harddiskhandlung integriert (backup) 45000 Adressen auf Harddisk 1800 pro 720 K Disk. **95,-**

Kontenführer: Die Finanzbuchhaltung unter GEM für professionelle Ansprüche **498,-**

Interprint: Universeller Druckertreiber. Integrierter Druckerspooler **49,-**

CAD: Bedienung über GEM und Makros Demo **25,-** Programm **998,-**

Haphazard: Grafikadventure mit hohem Spielwitz. Monochrom und Farbe **95,-**

WRITE 90°
- dreht Ihren Ausdruck um 90° (für überbreite Ausdrucke geeignet) **79,-**

TYPESETTER ELITE
Machen Sie Ihr eigenes Seitenlayout!
- bindet Text und Grafik mit ein - druckt mit 960x672 Pixeln - liest fremde Fonts **139,-**

MEGAFONT ST
- liest Ihr 1st Word Text File (und andere) und druckt es in verschiedenen Fonts - Grafikeinbindung möglich - eigene Fonts erstellbar **89,-**

Graphik
1.	Animator	119,-
2.	Maps and Legends	198,-
3.	Degas	169,-
4.	Easy-Draw	398,-
5.	Mica	298,-
6.	ST-Draw	448,-
7.	ST-Colouring (Neochrombilder)	125,-
8.	Platine ST	698,-
9.	Profi Painter	99,-
10.	Paintworks	98,-
11.	Grafic Artist	698,-

Drucker
Star SG10	698,-
NL 10 mit Interface	898,-
Panasonic 1080	798,-
Panasonic 1091	998,-
Panasonic 1092	1098,-
Okidata ML 182	898,-
Okidata ML 192	1398,-
C-Itoh Riteman F+	898,-
Epson LX80	998,-
Epson FX85	1278,-
Epson FX86	898,-
Okimate 20	848,-
SMM 804	698,-

TELEFONISCHE BESTELLUNG: 07191/15 28/29

Anwendungen

*Fußnotenverwalt. zu 1st Word	79,-
1-st-Word / 99,- deutsch	149,-
SM-Text (Sonderpreis)	180,-
Textomat ST	99,-
Datamat ST	99,-
DB-Master one	99,-
Text-Design ST	99,-
Profimat ST	99,-
DB-Man (Datenbank)	385,-
Trimbase (Datenbank)	298,-
Megabase	498,-
Format Copy V.1,4	89,-
Side-Click (Uhr, Termin Rechner)	145,-
BS-Handel (Bericht ST-7, Juli/August)	948,-
VT100 Terminal Emulator	195,-

Literatur

Für Einsteiger	29,-
Tips und Tricks	49,-
Peeks und Pokes	29,-
von Basic zu C	39,-
Das CP/M Buch	39,-
3D Grafik und Programm	59,-
GEM Draw, Degas, Neochr.	29,-
C-Programmierung/TOS	52,-
Atari ST Logo-Handbuch	49,-

Neueste Spiele:

1.	Shanghai	95,-
2.	Baslutball	95,-
3.	Juwels of Darkness	89,-
4.	Tee up (Golf)	59,-
5.	Super Huey	
6.	Electronic Pool	79,-
7.	The Pinball	79,-
8.	Strip Poker	79,-
9.	Q-Ball	79,-
10.	Time Blast	59,-
11.	Tass Times	95,-
12.	Space Station	95,-
13.	Karate	95,-
14.	Protector	95,-
15.	A Mind Forever Voaging	89,-
16.	Cards	79,-
17.	Infidel	89,-
18.	Suspect	89,-
19.	Sidewinder	89,-
20.	Mercenary	89,-
21.	Leather Goddesses	89,-
22.	Psion Chess	98,-

PUBLIC-DOMAIN-SOFTWARE

Wir liefern auf erstklassigem Diskettenmaterial die PD-Software 1-35 und weitere (siehe PD-Liste in dieser Ausgabe) zu **je 12,- DM.**

Ebenfalls erhältlich: Sämtliche in ST-Computer veröffentlichte Programme auf Diskette:

Jan-Feb; März-Apr; Mai-Juni; Sept-Okt; Nov-Dez **je 28,- DM**
Jul-Aug **18,- DM**

Ferner haben wir für Sie zusammengetragen:

100. Diskettenverwaltungsprogramm
101. Digimusik Oxygen
102. TOS vom 2.6.86
103. Basicprogramme, Terminalprogramm
104. Pyramide, ST-Pic, Zeichenprogramm
105. Finanzprogramm, Musikprogramm
106. Filecopy, Convertierprogramm für Mon.
107. Diskmonitor
108. Fonts, KermitGEM
109. Sounds, Analoguhr, Calc
110. Dirprint, Diskmon, Ramcopy
111. Haufenweise Accessories
112. Druckertreiber für 1st-Word
113. Sounds, Spiele
114. Copyram, Diskmonitor, Drucker
115. Mauseditor, Druckertreiber
116. Diskmonitor, Spooler, CommandTos
117. Bilder, Druckereinstellung STAR
118. Profi-Painter Demo
119. Digimusik: Foreign Affaire
120. Grafikprogramme haufenweise
121. Schach, Comiks
122. Ramdisk, ST-GRAFIK
123. Mauseditor, GEM-Draw
124. Logo, verschiedene Basicprogr.
125. Puzzle, Mramdisk, Schach, Text
126. Druckertreiber für Panasonic u.a.
127. Degasfonts, Formatierprogramm
128. Moire-Bilder, Liniengrafik
129. wieder viele Accessories
130. Grafikdemo

Weitere PD in Vorbereitung. Ausführlicher Katalog wird bei einer Bestellung mitgeliefert. Wir haben ständig gute Public-Domain-Software!

Metacomco Software

Hochwertige Entwicklungspakete zum günstigen Preis!

Lattice C, der Compiler	**348,-**
Makro-Assembler	**168,-**
Pascal-Compiler	**248,-**
NEU: (Beschreibung in ST 11/86)	
Make	**168,-**
Lisp	**498,-**
BCPL-Compiler	**329,-**

Disk-Royal **89,- DM**	Komfortabler Disk-Monitor unter GEM
Katpro-Royal **89,- DM**	Diskettenverwaltung der Spitzenklasse
Wizard-Royal **139,- DM**	Logik-Abenteuer und Geschicklichkeitsspiel mit Gerne-Konstruktionsset
Voc-Royal **59,- DM**	Mausgesteuertes Vokabel-Lernprogramm (englisch, französisch, italienisch, spanisch, russisch, griechisch, Latein)
Ctask-Royal **89,- DM**	Multitasking Bibliothek unter TOS + GEM

Wichtiges Zubehör

Diskettenbox 3.5"/40	39,80
Diskettenbox 3.5"/80	46,80
Weide Erweiterung auf 1 MB (ohne Löten)	275,-
Eprommer Atari ST	348,-
Eprom-Karte	98,-
Steckplatzerweiterung	189,-
Pal-Interface (HF-Modulator)	298,-
Digitizer (digitale Graphik)	598,-
Disketten 1DD 10 Stück	49,-
Disketten 2DD 10 Stück	59,-
Schaltpläne für alle Atari	29,80
Druckerkabel	38,-
Druckerständer Rauchglas	98,-
Endlospapier 1000 Blatt	29,-
Monitor Dreh-Kipp-Fuß	48,-
Thomson-Farbmonitor 36512	898,-
Thomson-Farbmonitor 36382	1298,-
Orion-Farbmonitor	848,-
Einzelblatteinzug für NL 10	290,-
Farbband SG 10	9,80
Farbband NL 10, Oki	24,80
Farbband Epson, SMM 804	22,80
Farbband Oki 20, Riteman	19,80
Atari Trackball	98,-
CSF-Gehäuse für 260/520 ST	198,-
ROM-Satz für alle ST's	140,-
Div. Kabel u. Stecker a. A.	

Schaltplanservice

Für folgende Atari-Geräte:

Atari 260 ST / 520 ST	29,80
Atari 520 ST +	29,80
Atari 1040 STF	29,80
Atari 520 STM	29,80
Atari 314 SF	14,80
Atari 354 SF	14,80
Atari SNM 804	14,80
Atari 600	14,80
Atari 800	14,80
Atari 1050	14,80

NEUHEITEN NEUHEITEN

Neue M+T-Programme

Protext ST

Eine professionelle Textverarbeitung für den ST

- Vollautomatische Silbentrennung mit einstellbarem Trenngrad
- Blockoperationen, Textattribute
- Druckbild auf dem Bildschirm
- Maus unterstützt
- Schnelles Position Suchen

DM 148,-

Speichererweiterung

Noch nie war es so einfach, seinen Computer aufzurüsten, wie es jetzt durch die

Weide-Speichererweiterung

möglich ist. Müheloser Einbau nur durch Stecken auf 512 K.

DM 275,-

Neue M+T-Programme

d'base II ST

Nun gibt es endlich den Standard unter den Datenbanksprachen auch für den Atari ST. Es unterstützt die Schnelligkeit des 68000-Prozessors voll und ganz und läßt sich komfortabel über Pull-Down-Menüs mit der Maus steuern.

Programm **DM 348,-**
Buch zum Programm **DM 49,-**

Echtzeituhr für Atari ST

Einbau der Uhr durch Einstecken in den Rechner, ohne Löten, sehr einfach.

- Datum, Uhrzeit
- Uhrenbaustein ist rtc 58321 h
- Langzeit-Genauigkeit
- Lithium-Batterien, 10 Jahre
- Einbau problemlos
- Programm für Auto-Ordner
- Schaltjahrerkennung

nur **DM 129,-**

Copy Star V1.8

Das Programm, das jeden Kopierschutz ignoriert und somit erlaubt, von Ihrem geschützten Programm eine Sicherheitskopie zu machen. Sie erhalten eine Normalkopie in 34 sec.!!!

Auch die Konvertierung in ein schnelleres Spezialformat und eine Erweiterung der Diskettenkapazität um 50 bzw. 100 KB ist damit möglich.

DM 169,-

Programmiersprachen

Die Entwicklungspakete

GST-Makro-Assembler	149,-
GST-C-Compiler	295,-
MCC-Makro-Assembler	168,-
MCC-Pascal-Compiler	248,-
MCC-Lattice C	348,-
Fast-Basic-Compiler	348,-
Fast-C-Compiler	398,-
Fast-Fortran 77	789,-
Fast-Pascal-Compiler	490,-
Prospero-Fortran 77	488,-
Prospero-Pascal	448,-
Laser Base	298,-
UCSD-P System Pascal	349,-
Modula 2 ST Compiler	349,-
Basic M-Compiler	398,-
Lisp-Compiler/Interpreter	490,-
4x Forth Level 1 (Superschnell)	348,-
4x Forth Level 2 (Mit Gem)	548,-
Forth ST	99,-
K-Seka Assembler	148,-

Atari-Trackball

Der Trackball als Alternative zur Maus, geringer Platzbedarf, gute Positionierung, mit Umbauanleitung **DM 98,-**
Anschlußfertig umgebaut **DM 128,-**

Marconi-Trackball

Höchste Präzision!!
Wurde in England für den Militärbereich entwickelt (Zielflugeinrichtung, Raketensteuerung, Panzerabwehr) bestens geeignet für CAM/CAD. Profigerät für Profianwender.
Anschlußfertig **DM 198,-**

Modula 2 ST

Modula 2 ist die Supersprache, die Ihnen alle Möglichkeiten von Pascal bietet und als besondere Feature das modulare Konzept, das besonders dem kommerziellen Programmierer die Arbeit wesentlich erleichtert.

- Volle GEM-Unterstützung
- Schnelles Laufverhalten
- Gute Dokumentation
- Sehr gute Ram-Disc wird mitgeliefert

DM 349,-

Hiermit bestelle ich per Nachnahme (Versandkostenpauschale 3,80/6,80)

Vorname, Name

Straße, Hausnummer

PLZ, Ort

Telefon-Nummer

Datum, Unterschrift

ST Aided Design – kurz STAD

– mehr als nur ein Zeichenprogramm!

Wer die Public Domain Diskette Nr. 5 der ST-Computer-Redaktion kennt, dem ist sicher ein kleines Demonstrationsprogramm namens ST Grafic aufgefallen. Es zeigt, was man zeichnerisch alles mit einem ATARI ST und der richtigen Software anfangen kann. Das Programm, mit dem diese kleine Demo erstellt wurde, lag uns nun in der Version 1.0 zum Test vor.

STAD ist ein monochromes 2D-Zeichenprogramm, dessen Vielfalt an Funktionen kaum einen Wunsch offen läßt. Zu diesem Teil gehört übrigens auch noch ein eigener Zeichensatz-Editor. Aber STAD ist auch ein 3D-Konstruktionsprogramm, mit dem sich Rotationskörper und Drahtgittermodelle erstellen und in Echtzeit im Raum drehen lassen. Der Clou dabei ist, daß sich Zeichnungen oder 3D-Körper problemlos zwischen den beiden Bereichen austauschen lassen. Man kann also mit dem 2D-Teil ein Objekt zeichnen, es dann im 3D-Teil perspektivisch verfremden und anschließend wieder zurück in den 2D-Teil übernehmen.

Das Programm wurde fast vollständig – bis auf einen kleinen Teil, der in C geschrieben ist – in Assembler erstellt. Das macht sich vor allem in der Geschwindigkeit der einzelnen Routinen bemerkbar, die der Programmautor Peter Melzer selbst geschrieben und optimiert hat. Alle Vorgänge laufen in Echtzeit ab, so daß man keine längeren Kaffeepausen einlegen muß, um die Umrechnungen abzuwarten, wie es bei so manchen auf dem Markt befindlichen Programmen üblich ist. Die einzelnen Programmteile werden nun ausführlich besprochen. Doch manchmal sagen Bilder mehr als Worte.

Zeichnen mit dem zweidimensionalen Teil

Neben den fast schon standardisierten Funktionen eines Zeichenprogramms

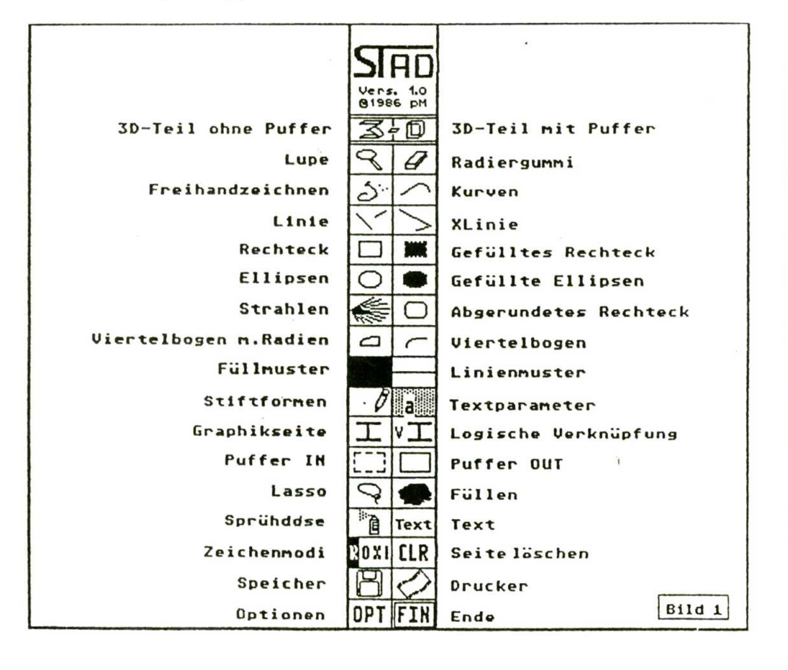


Bild 1: Das Menü des Zeichenteils

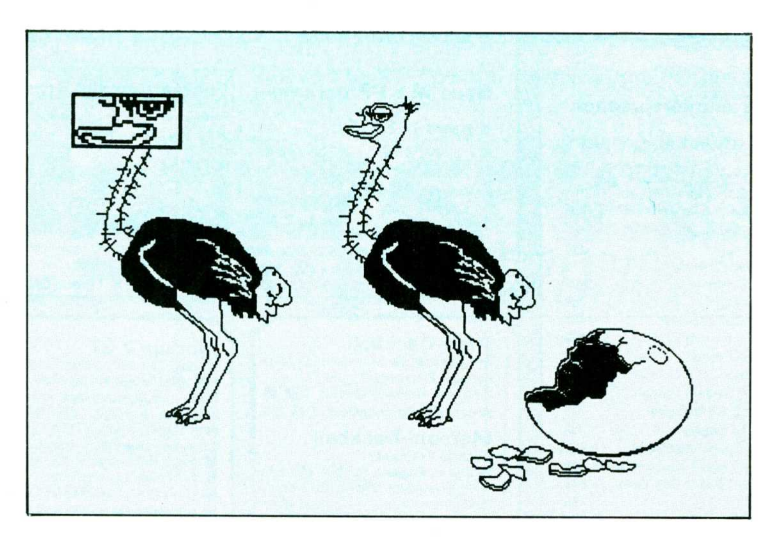


Bild 2: Suchen mit der Lupe

bietet STAD noch eine Menge zusätzlicher Möglichkeiten. Wenn man sich Bild 1 ansieht, wird man zuerst denken, daß man ein ganz normales Zeichenprogramm vor sich hat. Doch fast jede durch ein Symbol beschriebene Funktion bietet weitere Möglichkeiten, die sich auf den ersten Blick nur vermuten lassen. Alle Funktionen wurden bis zur Perfektion getrieben. So kann man natürlich alle Linien- und Füllmuster, die Sprühdose, den Cursor, die Zeichensätze, die Stiftformen und sogar den Radiergummi luxuriös editieren.

Jede einzelne Funktion ist mit Liebe zum Detail programmiert. So erhält man nach Aufruf der Lupe auch wirklich eine Lupe, die den optischen Gesetzen entspricht (Bild 2) und mit der man den gewünschten Zeichnungsteil vergrößern und bearbeiten kann (Bild 3). Schön ist auch die Funktion „Kurven", mit der man – wie sollte es auch anders sein – Kurven zeichnen kann. Nach Aktivierung der Funktion erhält man eine Gerade, die nun an beliebiger Stelle zur Kurve geformt werden kann. Und das in Echtzeit! Mit der „Lasso"-Funktion lassen sich beliebige Segmente durch Umfahren der Konturen aus einer Zeichnung herausschneiden und anschließend weiterverarbeiten.

Fünf verschiedene Zeichensätze gleichzeitig

Durch das Feld Textparameter gelangt man in das Auswahlmenü, das auf Bild 4 zu sehen ist. Dort stehen alle von GEM gewohnten Möglichkeiten zur Veränderung der Schriftarten zur freien Verfügung. Sie werden zum direkten Vergleich in einer Zeile im oberen Teil des Menüs angezeigt. Zudem lassen sich die Texte 90°-weise drehen. Durch Anklicken von Load kann man andere Zeichensätze nachladen und bis zu fünf verschiedene resident im Speicher halten. Klickt man Edit an, wird der Zeichensatzeditor nachgeladen (Bild 5), indem man zwei Zeichensätze gleichzeitig bearbeiten kann. Auf einfache Art und Weise kann man dort Zeichen entwerfen und zur späteren Verwendung abspeichern. Eine Auswahl von Zeichensätzen wird natürlich mitgeliefert.

Mehrere Graphikseiten

STAD verfügt über maximal 15 Graphikseiten, von denen eine als Graphikpuffer dient. Die Anzahl hängt vom freien Speicherplatz ab. Das Programm prüft automatisch, wieviele Graphikseiten in den Speicher passen, und läßt dann nur diese Anzahl als Maximum zu. Beim Laden von STAD kann indes die Anzahl der Graphikseiten auch selbst bestimmt werden. Zwischen den einzelnen Seiten können mit Hilfe des Puffers oder mit einer logischen ODER-Verknüpfung beliebige Bildschirminhalte ausgetauscht

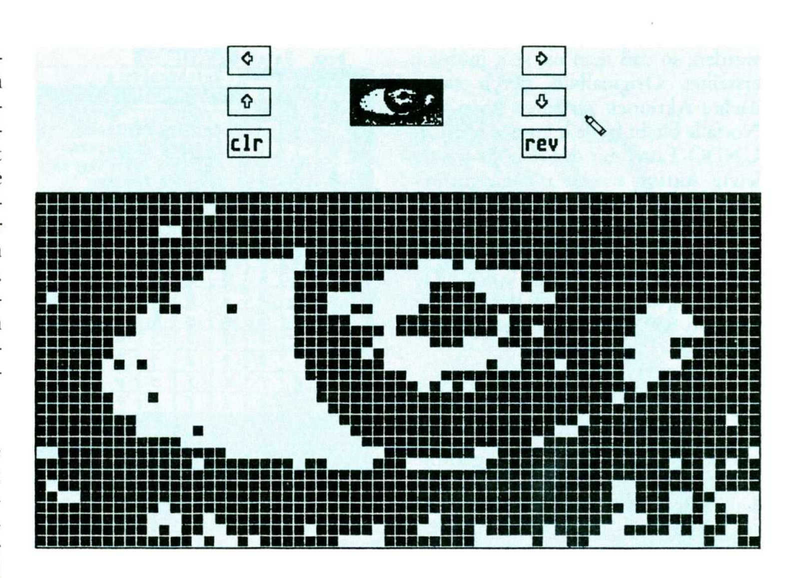


Bild 3: Manipulieren mit der Lupe

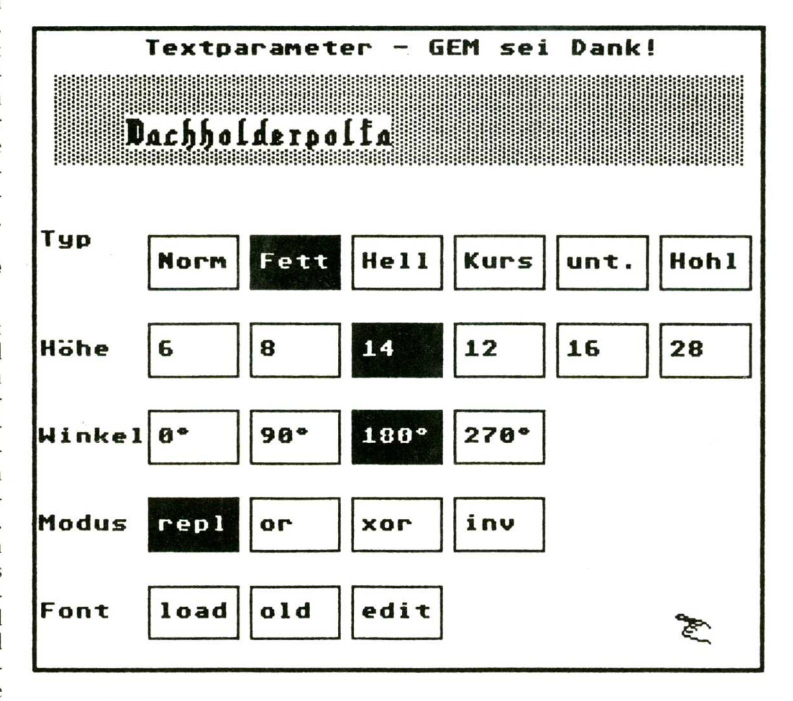


Bild 4: Das Textparametermenü

werden, so daß man nie sein mühsam erstelltes Originalbild durch unbedachte Aktionen zerstören kann. Für Notfälle bleibt freilich immer noch die UNDO-Taste, mit der sich die jeweilt letzte Aktion wieder rückgängig machen läßt. STAD sieht, wie oben bereits angedeutet, vor, daß sich verschiedene Graphikseiten logisch miteinander verknüpfen lassen. Zur Verfügung stehen dabei ODER-, UND-, Exklusiv ODER-Verknüpfungen und Negation.

Die beiden Pufferfunktionen von STAD bieten ebenfalls eine Vielzahl von Bearbeitungsmöglichkeiten. Es können zuvor definierte Bildausschnitte oder auch die ganze Graphikseite in den Puffer übernommen werden und anschließend u. a. verschoben, kopiert oder in der Struktur verändert werden. Bild 6 zeigt das einblendbare Helpmenü mit allen Pufferoperationen.

Bild 5: Der Zeichensatzeditor

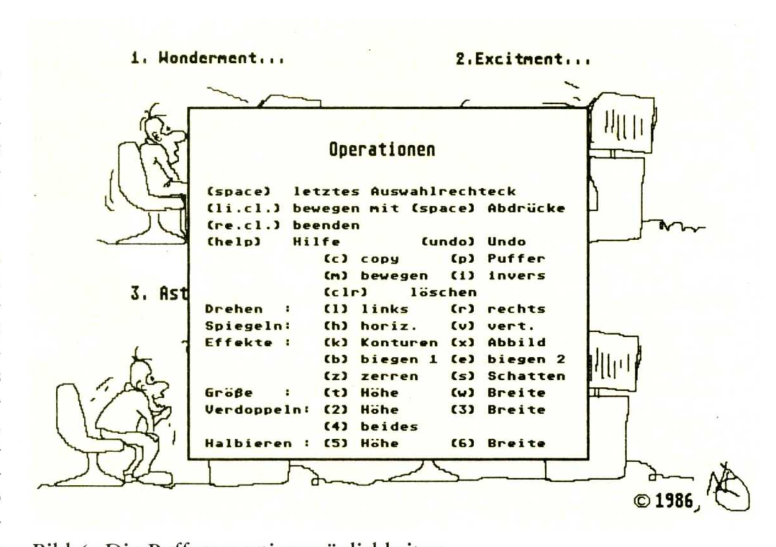


Bild 6: Die Pufferoperationsmöglichkeiten

Floppy, Scanner, Drucker...

Auch beim Zugriff auf die Peripherie wurden bei STAD alle Register gezogen. Laden lassen sich fast alle Bildformate der bekannten Zeichenprogramme (bis 64 KByte). Darunter fallen DEGAS (Elite), Neochrome, Mono- bzw. Colorstar, Profipainter und Doodle

Wie kann ein monochromes Programm Farbbilder laden? Die Antwort ist ganz einfach. STAD wandelt Farbbilder ohne großen Zeitverlust sofort beim Laden in qualitativ gute, monochrome Bilder um. Gegebenfalls sollte man die Graphikseite noch invertieren, falls das Bild zu hell wird. Beim Speichern stehen ebenfalls verschiedene Modi zur Verfügung. So lassen sich Graphikseiten gepackt, also platzsparend, im Doodle- oder DEGASformat, oder auch als Sequenz, d. h. mehrere Seiten auf einmal, abspeichern. Man kann die Graphikdaten außerdem auch als Assemblerdaten im Wortformat (dc.w) auf Diskette ablegen und somit in eigenen Programmen weiterverwenden.

STAD besitzt eine SHELL-Funktion, so daß man von STAD aus andere Programme laden und starten kann. Nach Beendigung dieser Programme kehrt man automatisch in STAD zurück und arbeitet wie gewohnt weiter.

Doch nicht nur von Diskette lassen sich Bilder in STAD holen. Es besteht auch die Möglichkeit, die Bilderdaten über die RS232-Schnittstelle zwischen dem ATARI ST und anderen Rechnern auszutauschen. Weiterhin wird im STAD-Handbuch eine Bauanleitung zu einem Scanner mitgeliefert, mit dessen Hilfe eine Bildervorlage eingelesen werden kann. Das Fotoelement des Scanners muß dazu auf dem Druckkopf eines Druckers befestigt werden. Die Bildqualität des Scanners können Sie auf Bild 7 beurteilen.

Nun zum Druckerteil. In der Defaulteinstellung ist STAD für EPSON-kompatible Drucker vorgesehen. Es wurde uns aber versichert, daß zur Zeit auch ein 24-Nadel Druckertreiber erstellt wird. Im Drucker-Menüteil ist ein leichtes Anpassen auf eventuell verschiedene Druckercodes möglich, so daß es mit dem Ausdruck im Grunde keine Probleme geben sollte. STAD bietet mehrere Druckformate an. So lassen sich z. B. bis zu vier Bildschirme auf ein Blatt Papier ausdrucken. Aber auch ein DIN A4-Format pro Graphikseite stellt keine große Schwierigkeit dar.

Optionen des 2D-Teils

Das Optionenmenü bildet die Ergänzung zu den bisher vorgestellten Möglichkeiten von STAD (siehe Bild 8). Hier lassen sich wichtige Voreinstellungen für die übrigen Funktionen auswählen. Man kann z. B. bis zu vier Graphikseiten auf einer Graphikseite verkleinert darstellen, um so die Übergänge zwischen den Seiten zu kontrollieren. STAD bietet außerdem eine Snap-Funktion und die Möglichkeit, selbstdefinierte Raster einzublenden, wodurch man punktgenau arbeiten kann. Ferner kann man das Symmetriezentrum neu bestimmen und Objekte um Achsen spiegeln. Mit den Tinies-Funktionen läßt sich z. B. erreichen, daß man Linien zeichnen kann, an deren Enden Pfeile erscheinen. Dies ist vor allem zum Bemaßen von Zeichnungen geeignet.

Ein besonderer Leckerbissen ist die Animation in STAD. Man kann die einzelnen Animationssequenzen auf den verschiedenen Graphikseiten (max. also 15) positionieren und anschließend mit variabler Geschwindigkeit ablaufen lassen. Damit lassen sich sehr schöne Minifilme mit gleitenden Bewegungsabläufen erstellen.

Schön ist auch, daß man die Hintergründe des 2D-Teils mit in den 3D-Teil übernehmen kann. Dabei kann z. B. als Hintergrund eine Wiese genauso wie ein Raster zum genaueren Zeichnen dienen. Der Hintergrund kann im 3D-Teil natürlich auch nachträglich gelöscht werden.

Objekte im dreidimensionalen Raum

Der 3D-Teil verfügt im Gegensatz zum 2D-Teil über Menüleisten. Man kann also Accessories verwenden (z. B. Snapshot, um die erstellten Bilder bzw. Objekte in 1st_Word plus zu bringen). Ebenfalls kann man aus dem 3D-Teil heraus Disketten formatieren, so daß man ruhig auch mal einen komplizierten Objektblock erstellen kann, ohne sich zuvor um ausreichenden Speicherplatz auf der Diskette zu kümmern.

Im dreidimensionalen Teil von STAD kann man, wie bereits oben erwähnt, relativ einfach Rotationskörper und Drahtgittermodelle erstellen. Es können mehrere 3D-Objekte (bis zu 100) zu einem Objektblock zusammengesetzt werden, wobei jedes Objekt ver-



Bild 7: Die Qualität des Scanners ist überzeugend

Optionen

one for	|two|	-two-	four	
Koordinaten	zeige	snap		
Symm.Zentrum	x= 319	y= 199		
Symm.Achse	keine	x-Ach	y-Ach	0-Pkt
Cursor	normal	neu	edit	Kreuz
Maus	normal	slower		
Sprühdose	normal	neu	edit	
Tinies	Rahm.	Umkr.	Radi	Pfeil
Aus-Puffer	repl	or	xor	and
Puffer	load	save	code	
Füllen	lokal	global		
Animation	los	az 015	wt 000	

Bild 8: Das Optionenmenü

schiedene Basiswerte, wie z. B. seine Lage im Raum oder den Standpunkt des Beobachters hat. Ein Objekt kann einzeln abgespeichert und später „hinzugemerget" werden. So kann man sich eine regelrechte Objektbibliothek anlegen. Auf Wunsch wird eine komplette Liste mit den Basiswerten der verwendeten Objekte ausgegeben.

Die erstellten Objektblöcke lassen sich um alle Achsen im Raum drehen, stufenlos vergrößeren, verkleinern und anschließend in der gewünschten Stellung fixieren und übernehmen. Auch hier ist die Geschwindigkeit, mit der die Objekte bewegt werden, erstaunlich. Alle Koordinaten (X, Y und Z) kann man vertauschen und so verschiedene Ansichten des Objektblocks erhalten. STAD bietet drei Projektionsarten zur Darstellung, nämlich die Zentral-, Parallel- und Normalprojektion. Zur besseren Orientierung läßt sich ein Achsenkreuz einblenden.

Ferner kann ein Hidden-Line-Modus hinzugeschaltet werden, damit verdeckte Kanten gelöscht werden. Hier ist auch die einzige Kritik anzusetzen: Der verwendete Algorithmus arbeitet nur bei konvexen Körpern zuverlässig und löscht bei anderen Körpern nicht immer alle Kanten. Eine Rückfrage beim Autor ergab jedoch, daß der 3D-Teil in Zukunft noch weitere wichtige Verbesserungen erhalten wird und der Käufer dann über einen Update-Service die neue Version erhalten kann. So kommt z. B. noch ein Schattierungsmodus hinzu, mit dem man den Lichteinfall aus verschiedenen Richtungen simulieren kann.

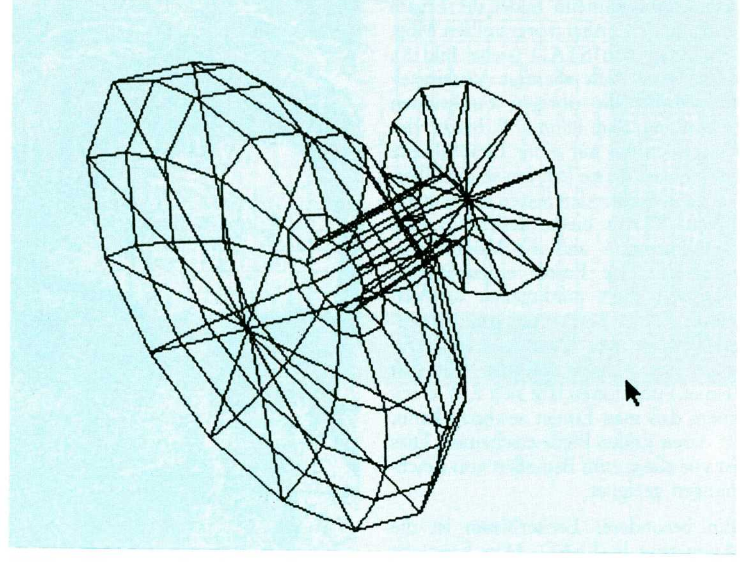


Bild 9: Ein Rotationskörper im 3D-Teil

Fazit

Mit STAD erhält sowohl der Zeichenkünstler als auch der Student, der Konstruktionszeichnungen erstellen will, ein sehr leistungsfähiges Programm, das wohl unter den monochromen Programmen dieser Kategorie seinesgleichen sucht. Es sei betont, daß es sich bei STAD nicht um ein CAD-Programm handelt, obwohl es schon nahe an diesen Breich herankommt. STAD ist über die Heidelberger Softwarefirma Application Systems zu beziehen und kostet DM 179,–.

PS: Wir haben STAD auch zusammen mit einem Graphiktablett der Königsteiner Firma KFC getestet und mußten feststellen, daß sich hiermit sehr genau arbeiten läßt, da STAD eigene Maus-Routinen benutzt (im Gegensatz zu anderen Zeichenprogrammen) und eine zusätzliche Option für Graphiktabletts verfügt. Allerdings sind diese Tabletts nicht ganz billig in der Anschaffung: Das von uns verwendete kostet DM 2 198,–.

(HE)

TAPEPRINT:

Perfekt bedruckte Cassettenhüllen

Endlich ist Schluß mit wüst beschrifteten Cassettenhüllen: TAPE-PRINT bedruckt vollständig ein Cassetten-Etikett mit Knickfalzkennung und seitlicher Beschriftung. Außerdem kann auf der Innenseite z. B. der eigene Name aufgedruckt werden: Ein unschätzbarer Vorteil bei Feten und Parties.



Im Prinzip handelt es sich um ein recht einfaches Programm. Das Ergebnis ist dennoch äußerst nützlich, denn TAPRPRINT druckt exakt im Format einer Cassettenhülle. Also: Titel eintippen, ausdrucken und in die Cassettenhülle schieben. Paßt.

Das Programm selbst ist in CCD-PASCAL geschrieben und dürfte auch ohne große Änderungen auf anderen Pascalsystemen laufen. Die hier verwendeten Druckersteuercodes sind die eines (hoffentlich) Epson-kompatiblen Gerätes. Für andere Druckertypen müssen lediglich die Steuercodes im Source-Listing angepaßt werden, z. B. der Zeilenvorschub beim NEC P6.

Die Bedienung ist sehr einfach. Nach dem Starten erscheint das Hauptmenü. Zuerst sollte „e“ für Eingabe gewählt werden. Es folgt die Eingabe der Daten für Seite A. Die Länge der Titel ist auf 28 Zeichen beschränkt, da eine Cassettenhülle leider nicht mehr Platz bietet. Aus dem gleichen Grund wurde die Zahl der Titel auf maximal 14 festgelegt.

Nach erfolgreicher Eingabe kann zur Überprüfung der Menüpunkt „Ausgabe“ gewählt werden, der alle Eingaben auf dem Bildschirm darstellt. Der Menüpunkt „Ausdrucken“ schließlich bringt die Daten zu Papier. Hinterher muß man das Blatt an den entsprechenden Stellen knicken und außenherum ausschneiden: Fertig ist die neue Hülle.

Karsten Wolf

```
program tapeprint (input,output);

var  seitea                     : array[0..15] of string[29];
     seiteb                     : array[0..15] of string[29];
     titela,titelb,namea,nameb : string[29];
     nummer                     : integer;
     ch                         : char;

procedure clear;
  begin
    write(chr(27),'E');
    writeln('      *************************');
    writeln('     *  T a p e - P r i n t  *');
    writeln('      *************************');
    writeln;
  end;

procedure drucken;
  const                                          | 65 mal - |
  leer='-----------------------------------------------------------------';
     name='Der Cassettenhuellenbedrucker von ST-Computer : TAPE-PRINT';
  var lauf:integer;
  begin
    rewrite(output,'prn:');
    write(chr(27),'@');                          | Drucker Reset |
    write(chr(27),'!',chr(4));                   | compressed    |
    writeln(leer);
    write(chr(27),'!',chr(24));                  | emphasized+double |
    writeln;
    writeln(' Seite A                  Seite B');
    write(chr(27),'S1',chr(27),'!',chr(4));     | subscript,compressed |
    write(chr(27),'1');                     | Zeilenabstand  7/72  |
   | write(chr(27),'3',chr(17)); z.B. NEC P6 Zeilenabstand 17/180 |
    writeln(leer);
    writeln(seitea[0],'       ',seiteb[0]);
    writeln(seitea[1],'       ',seiteb[1]);
    writeln(leer);writeln;
    for lauf:=2 to 15 do
      begin
        writeln(seitea[lauf],'  ',lauf-1:2,'  ',seiteb[lauf]);
      end;
    for lauf:=1 to 2 do writeln;
    writeln(leer);writeln;
    writeln('A: ',namea,'  "',titela,'"');
    writeln('B: ',nameb,'  "',titelb,'"');
    writeln;writeln(leer);
```

```
      for lauf:=1 to 4 do writeln;
      writeln(' ',name);
      writeln(leer);
      rewrite(output,'con:');
    end;

  procedure auffuellen;
    var lauf,laufl : integer;
    begin
      for lauf:=0 to 15 do
        begin
          for laufl:=length(seitea[lauf]) to 28 do
            begin
              seitea[lauf]:=concat(seitea[lauf],' ');
            end;
        end;
      for lauf:=0 to 15 do
        begin
          for laufl:=length(seiteb[lauf]) to 28 do
            begin
              seiteb[lauf]:=concat(seiteb[lauf],' ');
            end;
        end;
      if length(namea)<>length(nameb) then
        begin
          if length(namea)>length(nameb) then
            begin
              for lauf:=length(nameb) to length(namea)-1 do
                nameb:=concat(nameb,' ');
            end
          else
            begin
              for lauf:=length(namea) to length(nameb)-1 do
                namea:=concat(namea,' ');
            end;
        end;
    end;


  procedure eingeben;
    var lauf:integer;
    begin
      clear;
      writeln('     ---    Eingeben...     --- ');
      writeln;
      writeln('   SEITE A:     (alle angaben max. 28 Zeichen)');
      writeln('   Geben Sie den Namen des/der Interpreten ein ! ');
      readln(seitea[0]);
      writeln('   Geben Sie den Namen der Platte ein !');
      readln(seitea[1]);
      namea:=seitea[0];
      titela:=seitea[1];
      for lauf:=2 to 15 do
        begin
          write('   Titel Nr. ',lauf-1:2,' : ');
          readln(seitea[lauf]);
        end;
      readln;
      clear;
      writeln('     ---    Eingeben...     --- ');
      writeln;
      writeln('   SEITE B:     (alle angaben max. 28 Zeichen)');
      writeln('   Geben Sie den Namen des/der Interpreten ein ! ');
      readln(seiteb[0]);
      writeln('   Geben Sie den Namen der Platte ein !');
      readln(seiteb[1]);
      nameb:=seiteb[0];
      titelb:=seiteb[1];
      for lauf:=2 to 15 do
        begin
          write('   Titel Nr. ',lauf-1:2,' : ');
          readln(seiteb[lauf]);
        end;
      readln;
      auffuellen;
    end;

  procedure ausgeben;
    var lauf:integer;
        zeile:string[70];
    begin
      clear;                                                | 68 mal - |
   zeile:='--------------------------------------------------------------------
      writeln(zeile);
      writeln(seitea[0],'  "',seitea[1],'"');
      writeln(seiteb[0],'  "',seiteb[1],'"');
      writeln(zeile);
      for lauf:=2 to 15 do
        begin
          writeln(seitea[lauf],'     ',seiteb[lauf]);
        end;
      writeln(zeile);
      readln;
    end;

  begin
    repeat
      clear;
      writeln('  Hauptmenue  ');
      writeln;
      writeln('  wollen sie ...');
      writeln;
      writeln('                 eingeben (e)');
      writeln;
      writeln('                 ausgeben (a)');
      writeln;
      writeln('                 drucken  (d)');
      writeln;
      writeln('                 quit     (q)');
      writeln;
      write('      Ihre Wahl ... ');
      read(ch);
      if ch='e' then eingeben;
      if ch='a' then ausgeben;
      if ch='d' then drucken
    until ch='q';
  end.
```

BUCH- UND SOFTWARE-BESTELLKARTE

Ich/Wir bestelle(n)

Menge	Bestell-Nr.	Autor/Titel	Stückpreis	gesamt DM
		GfA-BASIC-BUCH	**49,–**	
		Programm-Diskette zum GfA-Buch	**39,–**	

☐ Scheck über DM________ ist beigefügt
☐ Per Nachnahme

Zuzüglich DM 5,– Versandkostenanteil.

Bitte beachten:
Es werden nur Festbestellungen berücksichtigt.
Eine Rückgabemöglichkeit besteht nicht.
Ausnahme nur bei Beschädigung.

Name

Vorname

Straße/Nr.

PLZ Ort

Datum

Unterschrift

ST-COMPUTER Kleinanzeigen-Auftrag

Bitte veröffentlichen Sie für mich folgende Kleinanzeige in der angekreuzten Rubrik

Biete an ☐ Hardware ☐ Software **Ich suche** ☐ Hardware ☐ Software ☐ **Tausch** ☐ **Kontakte** ☐ Verschiedenes ☐

30 Buchstaben je Standardzeile – incl. Satzzeichen und Wortzwischenräume.
Groß- und Kleinbuchstaben verwenden, fettgedruckte Wörter unterstreichen.

Bearbeitung **nur gegen Vorausscheck** über den entsprechenden Betrag (keine Überweisung)

☐ privat = DM 7,– je Zeile incl. MwSt.
☐ gewerblich = DM 15,– je Zeile + MwSt.
☐ Chiffregebühr = DM 10,–

☐ Scheck über DM________ ist beigefügt

Bei Angeboten: Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

Datum Unterschrift

Absenderangaben auf der Rückseite nicht vergessen

Kurzmitteilung

an die ST-Computer Redaktion

Zu dem Artikel ________________ in Heft____, Seite____ hätte ich folgendes zu bemerken:

☐ Ich möchte Ihnen folgendes Programm zur Veröffentlichung anbieten: (Kurzbeschreibung, Sprache, Länge in Druckerseiten, GEM/TOS)

☐ Ich kann über folgendes Thema berichten: (Tips & Tricks am ST, Hardware, Software, etc.)

☐ Ich möchte gerne Autor in der ST-Computer werden. Meine Fachgebiete: (z. B. LISP, Pearl, Modula-2, DBase, Assembler...)

☐ Ich möchte, daß folgendes Public-Domain Programm in Ihre Sammlung aufgenommen wird.

☐ Sonstiges

Bei weiteren Angaben oder Fragen wenden Sie sich bitte schriftlich oder telefonisch an die Redaktion. Tel. 06196/482158

ST-COMPUTER Abonnement

Ja, bitte senden Sie mir die ST-Computer Fachzeitschrift ab________
für mindestens 1 Jahr (11 Hefte) zum ermäßigten Preis von jährlich DM 60,– frei Haus.
(Ausland: Nur gegen Scheck-Voreinsendung DM 80,– Normalpost.)
Der Bezugszeitraum verlängert sich nur dann um ein Jahr, wenn nicht 6 Wochen vor Ablauf des Abonnements gekündigt wird.

Name

Vorname

Straße/Nr.

PLZ Ort

Gewünschte Zahlungsweise bitte ankreuzen

☐ Bequem und bargeldlos durch Bankeinzug

Konto-Nr. BLZ

Institut Ort

☐ Ein Verrechnungsscheck über DM_____ liegt bei.

☐ Gegen Rechnung

Garantie:
Diese Bestellung kann ich schriftlich innerhalb einer Woche (rechtzeitige Absendung genügt) widerrufen. Dies bestätige ich durch meine 2. Unterschrift.

Datum Unterschrift

Datum Unterschrift

Absender

Name

Vorname

Straße/Postfach

PLZ/Ort

Telefon:

Postkarte

Heim-Verlag
Heidelberger Landstraße 194

6100 Darmstadt-Eberstadt

Telefon 0 61 51 / 5 60 57

Absender

Name, Vorname

Straße

PLZ/Ort

Telefon

Postkarte

Heim-Verlag
ST-Computer
Redaktion
Schwalbacherstr. 64

6236 Eschborn

ST - Kleinanzeigen

Name

Vorname

Firma

Straße/Postfach

PLZ Ort

Datum Unterschrift

Bei Jugendlichen unter 18 Jahren bitte Unterschrift des Erziehungsberechtigten.

Postkarte

Bitte mit 60 Pf. frankieren

Anzeigenservice
Heim-VERLAG
Heidelberger Landstraße 194

6100 Darmstadt-Eberstadt

Telefon 0 61 51 / 5 53 75

ST ABO

Straße/Postfach

PLZ Ort

Telefon:

Postkarte

Bitte mit 60 Pf. frankieren

Heim-VERLAG
Heidelberger Landstraße 194

6100 Darmstadt-Eberstadt

Telefon 0 61 51 / 5 53 75

1000 Berlin



Kurfürstendamm 121a, 1000 Berlin 31 (Halensee)
Telefon 030/891 1082

Computare
Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d

DATAPLAY
Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61

Digital-Computer
Knesebeckstr. 76 · 1000 Berlin 12
Telefon
030-8 82 77 91







Vertragshändler
UNION ZEISS
Kurfürstendamm 57 • 1000 Berlin 15
Telefon 32 30 61

1000 Berlin



Keithstraße 26 · 1000 Berlin 30
☎ 26 111 26

2000 Hamburg

Gerhard u. Bernd Waller GbR
Computer & Zubehör
Kieler Straße 689
2000 Hamburg 54
☎ 040/570 60 07 + 570 52 75

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/49 44 00

Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon Sa. Nr. 040/641 50 91

2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt
Heiligengeiststr. 20, 2120 Lüneburg
Tel. 04131 / 461 22, Btx 40 24 22
Mo.-Fr. 9⁰⁰-18⁰⁰ und Sa. 9⁰⁰-13⁰⁰

2160 Stade

BERGAU
Büromaschinen · EDV-Systeme
Neue Straße 5, 2160 Stade
Telefon: (04141) 23 64 + 23 84

2210 Itzehoe

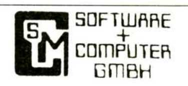
Der Computerladen
Inhaber Ulrich Bübel Martin Kopplow
Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90/91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · ☎ 04 31 / 56 70 42

2350 Neumünster



Klosterstraße 2 · 2350 Neumünster
Telefon (0 43 21) 4 39 33

2390 Flensburg

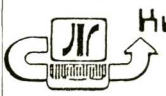


2800 Bremen

PS-DATA
Doventorsteinweg 41
2800 Bremen
Telefon 04 21 - 17 05 77

2850 Bremerhaven

HEIM- UND PERSONALCOMPUTER



Georgstraße 71
2850 Bremerhaven
Tel. 04 71 / 30 21 29
HARDWARE · SOFTWARE · PAPIERWARE

Das ATARI ST-Computer Sonderheft Nr. 1 gibt es jetzt auch bei Ihrem ATARI-Fachhändler.

2900 Oldenburg

wöltje
COMPUTER-CENTER
Atari · Schneider · Commodore
Multitech · Cumana · Taxan
2900 Oldenburg · Heiligengeiststr. 6
Tel. 04 41 - 40 45 89

2940 Wilhelmshaven

Radio Tiemann GmbH & Co. KG
2940 Wilhelmshaven
Telefon 0 44 21/2 61 45

2950 Leer



- HARDWARE-SOFTWARE
- EDV-SCHULUNG
- SYSTEM-ENTWICKLUNG
- EDV-BERATUNG
- ORGANISATION
- SERVICE-WARTUNG

Augustenstraße 3 · 2950 Leer
Telefon 04 91 - 45 89

3000 Hannover

Pro-Computer Hannover
Inh. HELGA PROSCHEK
• Beratung • Verkauf
• Programmierung • Installation • Service
☎ 05 11 / 52 25 79
D-3000 Hannover 71 · Großer Hillen 6



DATALOGIC COMPUTERSYSTEME
ATARI ST-COMPUTER HARDWARE SOFTWARE
BERATUNG SERVICE VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL. 0511 - 32 64 89

ATARI ST-Computer Sonderheft Nr. 1:
über 180 Seiten, jetzt bei Ihrem Fachhändler.

COM DATA
Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36

trendDATA Computer
IBM · EPSON · TRIUMPH ADLER
HEWLETT PACKARD · ATARI etc.
trendDATA Computer GmbH
Am Marstall 18-22 · 3000 Hannover 1
Telefon (05 11) 1 66 05-0

3040 Soltau

F & T Computervertrieb
Am Hornberg 1
(Industriegeb. Almhöhe)
3040 Soltau
Tel. 0 51 91 / 1 65 22

3100 Celle

Ludwig Haupt jr.
Büro-Einkaufs-Zentrum
Gerhard-Kamm-Straße 2
Ruf 8 30 45, Postfach 140
3100 Celle

3170 Gifhorn

COMPUTER-HAUS GIFHORN
Braunschweigerstr. 50
3170 Gifhorn
Telefon 0 53 71 - 5 44 98

3300 Braunschweig

COMPUTER STUDIO BRAUNSCHWEIG
Rebenring 49-50
3300 Braunschweig
Tel. (05 31) 33 32 77 / 78

3400 Göttingen



3400 Göttingen-Weende
Wagenstieg 14 — Tel. 05 51 / 3 40 31

3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler
Rudolf-Schwander-Str. 5, 9 + 13
3500 Kassel
Tel. (05 61) 70 00 00

3550 Marburg

L W M
COMPUTER SERVICE
Bahnhofstraße 26b
3550 Marburg/Lahn
✆ 0 64 21 - 6 22 36

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik — Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

4000 Düsseldorf

HOCO
EDV ANLAGEN GMBH
Flügelstr. 47
4000 Düsseldorf
Tel. 02 11 - 77 62 70

4050 Mönchengladbach



Hindenburgstr. 249
4050 Mönchengladbach
Tel. 0 21 61 - 18 7 64

4200 Oberhausen

KAMP
Büro- und Computersysteme
Vestische Straße 89/91
4200 Oberhausen 12 (Osterfeld)
Fernruf (02 08) 89 00 86
Fernschreiber 8 56 578

4290 Bocholt

Systemlösungen
A. W.-Karlen
Alfred-Flender-Str. 284 · 4290 Bocholt
Tel. 0 28 71 - 18 34 89
OKIDATA-Fachhändler

4300 Essen

Computerservice
Keese
Haus-Berge-Straße 163 · 4300 Essen 11
☎ (02 01) 66 70 62

ATARI Systemfachhändler


KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (02 01) 17 63 99

4400 Münster



COMPUTER SYSTEME GMBH
Daimlerweg 39 - 4400 Münster
Telefon 02 51 / 71 99 75 - 9

4422 Ahaus

ATARI · Epson · Fujitsu Molecular · NCR · Tandon · Schneider · Star **OCB**

OCB-Computershop
Wallstraße 3
4422 Ahaus
Tel. 0 25 61/50 21

OCB-Hard- und Software
Wessumerstraße 49
4422 Ahaus
Tel. 0 25 61/50 21

4430 Steinfurt

ATARI SCHNEIDER STAR EPSON

Computer
Büromaschinen
Service
4430 Steinfurt · Tecklenburger Str. 27
Telefon: 0 25 51 / 25 55

4500 Osnabrück

Heinicke-Electronic
Kommenderiestr. 120·4500 Osnabrück
Telefon 05 41 - 8 27 99
Wir liefern Micro-Computer seit 1978

4600 Dortmund

Bürostudio BOLZ
Brauhausstraße 4 · 4600 Dortmund
Telefon 02 31-52 77 13-16



Atari, Genie, Schneider, Tandy, Brother, Star, Memorex, BASF, Verbatim
cc Computer Studio GmbH
Software-Hardware-Beratung
Service-Eilversand
Ihre Ansprechpartner: v. Schablinski, Jan P. Schneider
Elisabethstraße 5, 4600 Dortmund 1
T. 0231/528184 · Tx 822631 cccsd

City Elektronic
Güntherstraße 75
4600 Dortmund
Telefon 02 31/57 22 84



4700 Hamm



4790 Paderborn



GESELLSCHAFT FÜR ELEKTRONISCHE TELEKOMMUNIKATION
IM SCHILDERN 15, 4790 PADERBORN
TEL. (0 52 51) 2 60 41
BTX ★51051#

4800 Bielefeld

hardware software organisation service **CSF**

CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106–108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

Carl-Severing-Str. 190
4800 Bielefeld 14
Telefon: 05 21/45 99-150
Telex : 9 37 340 krab d
Telefax: 05 21/45 99-123

MICROTEC
Software
Hardware
Beratung
Service

ST-Computer Einkaufsführer:
Werbewirksam, aktuell und preiswert

4930 Detmold

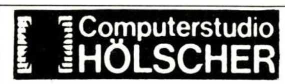
Frevert-Computer
Autorisierter Atari System-Händler
4930 Detmold
Bismarckstr. 12 **Tel. 0 52 31-2 24 16**
4920 Lemgo
Braker Mitte 9 **Tel. 0 52 61-8 85 20**, **8 84 40**
Autorisierter Commodore-Systemhändler für Lippe, Minden-Lübecke u. Herford

5000 Köln



5000 KÖLN 1
RICHARD-WAGNER-STR. 39
RUF: 02 21/21 91 71

5010 Bergheim



EDV-Beratung · Organisation
Programmierung · Home/Personal-Computer
Software · Zubehör · Fachliteratur
Zeppelinstr. 7 · 5010 Bergheim
Telefon 0 22 71-6 20 96

5060 Bergisch-Gladbach

Computer Center
Buchholzstraße 1
5060 Bergisch-Gladbach
Telefon 0 22 02-3 50 53

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5200 Siebgurg

Computer Center
Luisenstraße 26
5200 Siegburg
Telefon 0 22 41/6 68 54

5222 Morsbach

Computersysteme von A – Z
multicomp
Postanschrift: s. Hofmeister
Talstr. 22 · 5222 Morsbach-Wallerhausen
Telefon: 0 22 94 / 73 08

5400 Koblenz

SCHMITT COMPUTERSYSTEME
Casinostraße 40
5400 Koblenz
✆ 02 61-3 65 28

5412 Ransbach

Computer Technik Kieckbusch GmbH
Der Softwarespezialist
Am Seeufer 11+22 · 5412 Ransbach
Telefon 0 26 23-16 18

5457 Straßenhaus

DR. AUMANN GMBH
Computer-Systeme
Schulstr. 12
5457 Straßenhaus
Telefon 0 26 34 - 40 81/2

5500 Trier



Güterstr. 82 - 5500 Trier
☎ 06 51 - 2 50 44
Fordern Sie unsere Zubehör-Liste an.

5540 Prüm

ATC COMPUTER
J. ZABELL
Kalvarienbergstr. 34
5540 PRÜM
– Tel.: 0 65 51 - 34 83 –

5600 Wuppertal

Jung am Wall
Wall 31—33
5600 Wuppertal 1
Telefon 02 02/45 03 30

5630 Remscheid

C O M SOFT
Scheiderstr. 12 · 5630 Remscheid
Telefon (0 21 91) 2 10 33 - 34

5800 Hagen



Vertragshändler **Axel Böckem**
Computer + Textsysteme
Elper Str. 60 (Eilpezentrum) · 5800 Hagen
Tel. 0 23 31 / 7 34 90

5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

6000 Frankfurt

Müller & Nemecek
Kaiserstraße 44
6000 Frankfurt/M.
Tel. 0 69 - 23 25 44

WAIZENEGGER
Büroeinrichtungen
Kaiserstraße 41
6000 Frankfurt/M.
☎ 0 69 / 720 30 60

ATARI
... wir machen Spitzentechnologie preiswert.
Jetzt bei uns. Vertragshändler
bco
Büro-Computer + Organisations GmbH
Oederweg 7–9
6000 Frankfurt/M. 1
☎ (0 69) 55 04 56-57
Hardware ★ Software ★ Beratung ★ Service

SCHMITT
COMPUTERSYSTEME
Grosse Friedbergerstr. 30
6000 Frankfurt
✆ 0 69 - 28 40 65

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
✆ 0 61 51 / 5 60 57

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Elisabethenstr. 15 · 6100 Darmstadt
Luisencenter · Tel. 0 61 51 - 10 94 20

SCHMITT
COMPUTERSYSTEME
Mühlstraße 76
6100 Darmstadt
✆ 0 61 51 - 2 45 74

6200 Wiesbaden

SCHMITT
COMPUTERSYSTEME
Rheinstraße 41
6200 Wiesbaden
✆ 0 61 21 - 30 73 30

6240 Königstein

KFC
COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6300 Gießen



Schneider ATARI Commodore
BAUMS
BÜRO · ORGANISATION
Bahnhofstr. 26 · 6300 Gießen
Telefon: 06 41 / 7 10 96

6330 Wetzlar



Fachmarkt
für
Computer u.
Unterhaltungs-
electronic in Wetzlar,
Einkaufszentrum Bahnhofstraße, Tel. (0 64 41) 4 85 66

6400 Fulda

Schneider ATARI Commodore
WEINRICH
BÜRO · ORGANISATION
Ronsbachstraße 32 · 6400 Fulda
Telefon: 06 61 / 4 92 - 0

6457 Maintal

Landolt-Computer
Beratung · Service · Verkauf · Leasing
Wingertstr. 112
6457 Maintal/Dörnigheim
Telefon 0 61 81 - 4 52 93

6500 Mainz

:ELPHOTEC
Computer Systeme
Ihr Atari Systemhändler mit eigenem Service-Center
Schießgartenstraße 7
6500 Mainz
Telefon 0 61 31 - 23 19 47

SCHMITT COMPUTERSYSTEME
Karmeliterplatz 4
6500 Mainz
✆ 0 61 31 - 23 42 23

6520 Worms

ORION
Computersysteme GmbH
Friedrichstraße 22
6 5 2 0 W O R M S
Tel. 0 62 41 / 67 57 - 67 58

6700 Ludwigshafen

MKV Computermarkt
Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

MKV Computermarkt
Gilgenstraße 4
6720 Speyer
Telefon 0 62 32 - 7 72 16

6730 Neustadt

Felten & Meier
Computersysteme
Atari + OKIDATA Fachhändler
Exterstr. 4 · 6730 Neustadt
Tel. 0 63 21 / 8 89 94

6750 Kaiserslautern

C.O.S.
COMPUTER ORG. GmbH
Karl-Marx-Straße 8
6750 Kaiserslautern
Telefon (06 31) 6 50 61 - 62

6750 Kaiserslautern



6800 Mannheim



Computersysteme + Textsysteme
6800 Mannheim 24
Casterfeldstraße 74-76
☎ (06 21) 85 00 40 · Teletex 6 211 912

Computer-Center
am Hauptbahnhof GmbH
L 14, 16-17
6800 Mannheim 1
Tel. (06 21) 2 09 83 / 84

6900 Heidelberg

JACOM COMPUTERWELT
Hardware · Software
Schulung · Service
Mönchhofstraße 3 · 6900 Heidelberg
Telefon 0 62 21 / 41 05 14 - 550

Heidelberger Computer-Center
Bahnhofstraße 1
6900 Heidelberg
Telefon 0 62 21 / 2 71 32

7000 Stuttgart



COMPUTERFACHHANDEL
Der Computer-spezialist
star ATARI olivetti NEC
BNT · Computerfachhandel GmbH
Marktstraße 48 · 7000 Stuttgart 50
Mailbox 07 11/55 83 92
Datex P 45400091120
Telefon 07 11/55 83 83
Service 07 11/55 83 91
Tx 05 1933 521 dmbox g.
ref: box: dmz: bnt

SCHMITT COMPUTERSYSTEME
Tübingerstr. 18
7000 Stuttgart

7022 L.-Echterdingen

Autorisierter ATARI-System-Fachhändler
ATARI® ST
Matrai computer
Matrai Computer GmbH
Bernhäuser Str. 8
7022 L.-Echterdingen
☎ (07 11) 79 70 49

7030 Böblingen

mca Computer Center
Sindelfinger Allee 1
7030 Böblingen
Tel. 0 70 31/22 36 18

7070 Schwäb. Gmünd



Schwerzerallee 23
7070 Schwäbisch Gmünd
Telefon 07171/5633

7100 Heilbronn

Unser Wissen ist Ihr Vorteil
Walliser & Co.
Mönchseestraße 99
7100 Heilbronn
Telefon 07131/60048

Computer-Welt


Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 03

7150 Backnang



7410 Reutlingen

Computer-Shop
Werner Brock
autorisierter Fachhändler f. ATARI, Schneider, Oki
Federnseestr. 17 · 7410 Reutlingen
Telefon: 0 71 21 / 3 42 87

7450 Hechingen

SRE
Gesellschaft für Datenverarbeitung mbh
Computer · Drucker
Zubehör · Fachliteratur
Schloßplatz 3 · 7450 Hechingen
Telefon 07471/14507

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis
HEIM + PC-COMPUTERMARKT
HARDWARE · SOFTWARE · LITERATUR
SCHEURER
ATARI COMMODORE CUMANA DATA-BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON
7475 Meßstetten 1 · Hauptstraße 10 · 07431/61280

7480 Sigmaringen



7500 Karlsruhe

papierhaus erhardt
Am Ludwigsplatz · 7500 Karlsruhe
Tel. 0721-23925

MKV Computermarkt
Rüppurer Straße 2d
7500 Karlsruhe
Telefon 0721-373071

7530 Pforzheim

DM Computer GmbH
Hard- & Software
Durlacherstr. 39
Tel. 07231-13939
7530 Pforzheim
Telex 783248

8000 München

schulz computer
Schillerstraße 22
8000 München 2
Telefon (089) 597339
Beratung · Verkauf · Kundendienst

7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk
In der Jeuch 3
7600 Offenburg
Telefon 0781/57974

7640 Kehl/Rhein



Badstrasse 12
Tel. 07807/822
Telex: 752913
7607 NEURIED 2
Filiale:
Hauptstrasse 44
Tel. 07851/1822
7640 KEHL/RHEIN
eigener Service · eigene Software
ELEKTRO-MÜNTZER GmbH

7700 Singen

U. MEIER
Computersysteme
7700 Singen-Htwl.
Am Posthalterswäldle 8
Telefon 07731-44211

7730 VS-Schwenningen

BUS BRAUCH & SAUTER COMPUTER TECHNIK
Villinger Straße 85
7730 VS-Schwenningen
Telefon 07720/38071-72

7750 Konstanz

ATARI ★ PC's ★ SCHNEIDER
computer - fachgeschäft
rösler
Rheingutstr. 1 · ☏ 07531-21832

7800 Freiburg

CDS EDV-Service GmbH
Windausstraße 2
7800 Freiburg
Tel. 0761-81047

computer aktuell
Südbadens kompetenter Computer-Partner.
Kaiser-Joseph-Str. 232
7800 Freiburg, Tel.: 0761/2180225

7890 Waldshut-Tiengen

hettler-data
service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 07751/3094

7900 Ulm

HARD AND SOFT COMPUTER GMBH
Ulms großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Herrenkellergasse 16 · 7900 Ulm/Donau
Telefon 0731/62699

COMPUTERSTUDIO
Büro & Datentechnik · 2x in Ulm
Claus Wecker
Hafenbad 18/1 + Frauenstr. 28
7900 Ulm/Do.
Telefon (0731) 28076

7950 Biberach

HARD AND SOFT COMPUTER GMBH
Biberachs großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Schulstraße 6 · Bei der VHF
7950 Biberach · Tel. 07351/12221

8000 München

ludwig
COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STR. 62L
EURO-INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089/3113066 · TELETEX 898341

SCHMITT COMPUTERSYSTEME
Arnulfstraße
8000 München

comp +carry
COMPAQ


Das Computer-Fachgeschäft
im HERTIE Kaufhaus
Hauptbahnhof und Schwabing
Telefon 595277/348054
Ingolstädter Straße 20-24 · 8 München 45 · 089/3597091
Ein Unternehmen der procomp-Gruppe

8032 Gräfeling



COMPUTER SYSTEME SCHULUNG
Am Haag 5
8032 Gräfelfing
Tel. 089-8545464,851043

8070 Ingolstadt

DREYER GMBH
Elektrotechnik
Manchinger Straße 125
8070 Ingolstadt
Tel. 0841/6590

S & S
Computer-Center GmbH
Kuperstraße 20 · 8070 Ingolstadt
Tel. 0841-32812

8120 Weilheim

Klement
Elektro-, Radio- und Fernseh-Center
Beleuchtungskörper · Schallplatten
Meisterbetrieb · Computer Fachhändler
8120 Weilheim · Admiral-Hipper-Straße 1
Geschäft ✆ 4500 · Kundendienst ✆ 4400
Interfunk-Fachgeschäft

8150 Holzkirchen

ATARI
Besuchen Sie uns!
Fordern Sie unseren Software-Katalog (520ST) an!
M
MÜNZENLOHER GMBH
Tölzer Straße 5
D-8150 Holzkirchen
Telefon: (08024) 1814

8170 Bad Tölz

Elektronik Center
Bad Tölz
Wachterstraße 3
8170 Bad Tölz
Telefon 08041/41565

8220 Traunstein

computer studio
BÜROMASCHINEN
cs
Ludwigstraße 3 · 8220 Traunstein
Stadtplatz 10 · Tel. 0861-14767 o. 3905

ATARI ST-Computer Sonderheft Nr. 1:
über 180 Seiten, jetzt bei Ihrem Fachhändler.

8300 Landshut

BÜRO-DALLMER
Altstadt 69
8300 Landshut
Telefon 0871/2106264

8330 Eggenfelden

Hot Space
Computer-Centrum
R. Lanfermann
Schellenbruckstraße 6
8330 Eggenfelden
Telefon 08721/6573
Altöttinger Straße 2
8265 Neuötting
Telefon 08671/71610

8400 Regensburg

C-SOFT GMBH
Programmentwicklung & Hardware
Holzfällerstraße 4
8400 Regensburg
Telefon 0941/83986

Zimmermann elektroland
8400 Regensburg
Dr.-Gessler-Str. 8
☎ 0941/95085
8390 Passau
Meraner-Str. 5
☎ 0851/51016

8490 Cham



Auf der Schanze 4 · 8490 Cham/Opf. · Telefon (09971) 9723

8500 Nürnberg



8500 Nürnberg

SCHMITT COMPUTERSYSTEME
Innere Laufer Gasse 29
8500 Nürnberg
✆ 0911-209717



Der ATARI - Systemfachhändler in Nürnberg
Softwareberatung und Entwicklung
Computer • Drucker • Zubehör • Fachliteratur
hib
HIB Computer GmbH
Äußere Bayreuther Str. 72
8500 Nürnberg 21
Tel.: 0911/515939

8520 Erlangen



ALPHATRON
COMPUTERSYSTEME & SOFTWARE ENGINEERING
marco hildebrandt
luitpoldstraße 22 · 8520 erlangen

Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 09131/42076



8520 Erlangen
Dresdener Str. 5
Friedrichstr. 9
Tel.: 09131/12010

ST-Computer Einkaufsführer:
Werbewirksam, aktuell und preiswert

8600 Bamberg



8670 Hof

COMPUTER-CENTER-BURGER
Spezialist für Personal- und Home-Computer, Programme
Zubehör, Beratung, Service
8670 Hof · Leimitzer Straße 11 · Telefon 09281/40075 Abt. Computer



8700 Würzburg

HALLER GMBH
Fachgeschäft für
Mikrocomputer
Büttnerstraße 29
8700 Würzburg
Tel. 09 31 / 1 67 05



Hardware · Software
Service · Schulung

computer center
am Dominikanerplatz
Ruf (09 31) 5 04 88

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 65 21 54

8753 Mömbris

COMPUTIQUE
DIE EXPERTEN FUR MICROCOMPTER
Im Kahltal Zentrum, 8752 Mombris
Telefon (0 60 29) 65 20 oder 14 10

ATARI 520 ST
APRICOT
IBM Komp.
Festplatten

Hardware
Software
Schulung

8900 Augsburg

Adolf & Schmoll Computer
Schwalbenstr. 1 · 8900 Augsburg
Tel. (08 21) 52 85 33 oder 52 80 87
Wir sind außerdem autorisierte Service-Fachwerkstatt für:
Schneider
commodore
ATARI

8900 Augsburg

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Bürgermeister-Fischer-Str. 6-10
8900 Augsburg · Tel. (08 21) 31 53 - 416

SCHMITT COMPUTERSYSTEME

Frauentorstr. 22
8900 Augsburg
✆ 08 21 - 15 42 68

ST-Computer Einkaufsführer:

Werbewirksam, aktuell und preiswert

Sprechen Sie mit uns:
Heim-Verlag
Tel.: 0 61 51 / 5 60 57

8940 Memmingen

EDV-Organisation
Hard- + Software
Manfred Schweizer KG
Benninger Str. 34, Tel. 0 83 31 / 1 22 20
8940 Memmingen

Österreich

A-1020 Wien

W + H.
Computerhandel Ges m.b.H.

Förstergasse 6/3/2 · 1020 Wien
Tel. 02 22 - 35 09 68

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H.

1040 Wien - Paniglgasse 18-20
Telefon 02 22 - 65 78 08, 65 88 93

A-8010 Graz



A-8010 Graz, Mandellstraße 23, Tel. (0 31 6) 70 28 40, 78 39 23
Tlx.: 03 2534 zupan a

Schweiz

CH-1700 Fribourg

Softy Hard's Computershop
Die ATARI ST Spezialisten

Grand Rue 42
CH-1700 Fribourg
Tel. 0041(0) 37 22 26 28

CH-2503 Biel

UE
URWA Electronic
Ihr ATARI ST Spezialist in der Schweiz.
0 32 / 25 45 53
Lindenweg 24, 2503 Biel

CH-3000 Bern

Computer Corner

S. Bazan
Könizstrasse 70
3008 Bern
✆ 031/25 57 37

Hardware
Software
Zubehör
Occasion

CH-3415 Hasle Rüegsau

COMPU-TRADE
Ihr ATARI Spezialist
Emmenstr. 16
CH-3415 Hasle-Rüegsau
✆ 0 34 / 61 45 93 auch abends bis 21.00 h
HARD- u. SOFTWARE · BERATUNG · EILVERSAND

Alle 12 bisher erschienenen Ausgaben gibt's natürlich bei Ihrem Fachhändler.

CH-4054 Basel

COMPUTER CENTER

COMPUTER SYSTEME

DIE ST-SPEZIALISTEN IN BASEL

Öffnungszeiten:

Di. – Fr. 9.30 – 12.30 / 14.00 – 18.30
Sa. 9.30 – 16.00
Mo. geschlossen

SYSAG

COMPUTERSCHULE

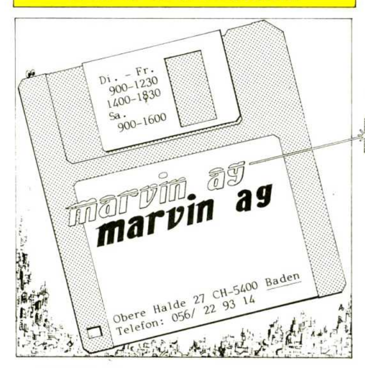
HOLEESTRASSE 87 · 4054 BASEL · TELEFON 061 39 25 25

CH-4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 0 62 / 63 17 27 + 63 10 27

CH-5400 Baden



CH-5430 Wettingen



CH-8006 Zürich

ADAG Computershop

Universitätsstr. 25 · 8006 Zürich
Tel. 01 / 47 35 54

ATARI & WANG & EPSON

CH-8006 Zürich

Computer-Center P. Fisch

Stampfenbachplatz 4
8006 ZÜRICH
✆ 01 / 363 67 67

CH-8021 Zürich

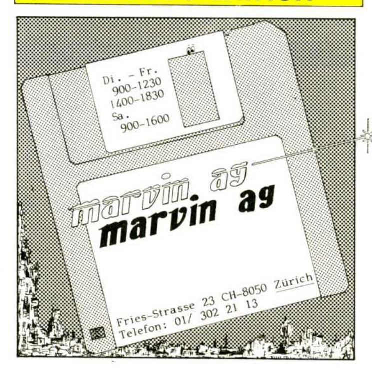


VILAN

Das Warenhaus der neuen Ideen

Ihr Computer-Fachhandel an der Bahnhofstrasse 75 · Zürich

CH-8050 Zürich



CH-8246 Langwiesen

ZIMELEC

Elektronik-Shop
8246 Langwiesen
☎ 0 53 55 22 4

Montag-Freitag von 10-20 Uhr geöffnet

Samstag von 10-16 Uhr.

CH-9000 St. Gallen

VIDEO - COMPUTER - CENTER
GÄCHTER AG

Webergasse 22
9000 St. Gallen
Telefon 071 / 22 60 05

video computer center

CH-9400 Rorschach

PAUS-electronic
Ihr Computer-Fachhandel
Industriestr. 30

CH-9400 Rorschach ✆ 071-41 18 85

CH-9450 Altstätten

PAUS-electronic

Ihr Computer-Fachhandel

Ringgasse 27 · CH-9450 Altstätten
Telefon 0 71 - 75 34 79

März '87 – Anzeigenschluß am 3. Februar 1987

April '87 – Anzeigenschluß am 3. März 1987

Verspätete Einsendungen kommen in die darauffolgende Ausgabe!

Luxemburg



GFA-DRAFT

CAD-Programme haben zur Zeit Hochkonjunktur auf dem ATARI ST. Ständig erscheinen neue auf dem Markt, so daß dieser immer unübersichtlicher wird. Auch die durch Ihr hervorragendes Basic bekannte Firma GfA-Systemtechnik, hat ein solches Programm entwickelt, daß besonders durch seinen niedrigen Preis von DM 298,– auffällt. CAD-Programme kosten normalerweise erheblich mehr – was kann man also von GFA-DRAFT erwarten.

GFA-DRAFT arbeitet mit der **Fenster- und Menütechnik**, fast alle Befehle werden mit der **Maus** abgesetzt. Diese nimmt somit die Funktion eines Grafiktabletts ein, wie es bei professionellen Anlagen eingesetzt wird. Durch die ständig angezeigten Koordinatenangaben (mm oder Zoll) wird dieses präzise, aber teuere Eingabegerät sehr gut ersetzt. Die Koordinaten können absolut oder relativ zum letzten Punkt angegeben werden. Sie können aber auch die Länge der Linie anzeigen, was besonders bei schrägen Linien sehr wichtig sit. Waagrechte und senkrechte Bewegungen lassen sich neben der Maus auch mit den Cursortasten ausführen, dabei sind große Sprünge ebenso möglich wie ein pixelweises Weiterrücken des Cursors. Der Cursor ist in der Grundeinstellung als großes Kreuz ausgeführt, dadurch lassen sich auch hiermit gut Waagrechte und Horizontale ziehen.

Der Zeichenbereich umfaßt normalerweise 658 x 658 mm, durch eine einfache **Maßstabsänderung** kann er jedoch in weiten Bereichen variiert werden. Der sichtbare Bildausschnitt beträgt meist nur einen Teil des Gesamten, aber auch er kann **'gezoomt'** werden um einen Überblick zu erhalten oder um Details zu bearbeiten.

Der Bildausschnitt wird mit den bekannten Scrollbalken über den Zeichenbereich geführt, wobei die Balken jedoch keinen guten Anhalt über die Lage geben.

Neben den Standardelementen Linien, Kreis, Kreissegment, Ellipse und Rechteck sind noch weitere **nützliche Konstruktionsbefehle** vorhanden:

- Kreisbogen durch drei Punkte
- Abrunden von Ecken
- Linien einrasten (wichtig für Schraffur)
- Vorzugsrichtung (hält Linien richtungsstabil)
- Linien knacken (teilt Linien an Schnittpunkten)
- Linien trimmen (schneidet überstehende Linien ab)
- Lot auf eine Gerade

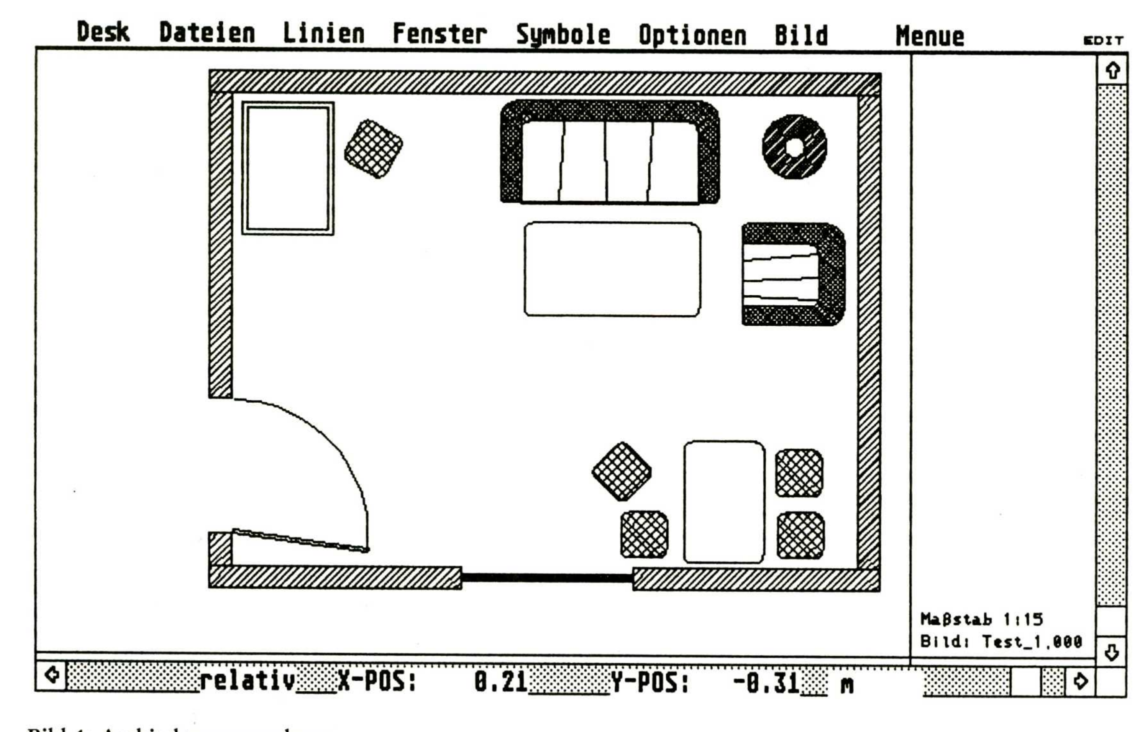


Bild 1: Architekturanwendung

- Gerade mit Winkel zu einer Ebene
- Freihand zeichnen

Eine der zeitaufwendigsten Zeichenarbeiten ist normalerweise das **Schraffieren von Flächen.** Auch diese Arbeit kann sauber und schnell vom Computer erledigt werden. Die Funktionen von GFA-DRAFT sind dazu sehr komfortabel. Nachdem eine Linie des gewünschten Körpers angewählt wurde, wird überprüft, ob die Linie zu einem völlig geschlossenen Gebilde gehört. Falls dies der Fall ist, wird die Umrißlinie hervorgehoben. Wenn nicht der richtige Körper getroffen wurde, kann die Suche auch fortgeführt werden.

Eine weitere aufwendige Arbeit ist das **Bemaßen der Zeichnung,** das jetzt jedoch sehr schnell geht. Dazu werden die beiden Linien angeklickt und schon erscheint die Bemaßungslinie mit Pfeilspitzen. Ein weiterer Tastendruck läßt dann sogar automatisch die Maßzahl erscheinen. Wenn die Linien nicht parallel sind, dann wird ein Radiusbogen mit der entsprechenden Winkelangabe gezogen.

Das Beschriften einer Zeichnung kann mit zwei Zeichensätzen und in **vier Richtungen** erfolgen. Sie kann auch **gespiegelt** werden falls ein Platinenlayout angefertigt wird. Die sieben Schriftgrößen sind nicht unbedingt normgerecht, denn die Größe wird immer verdoppelt.

Neben dem Bildausschnitt kann ein weiterer Bereich als **FENSTER** definiert werden. Darauf sind dann vielfältige Operationen anwendbar:

- ausschneiden
- löschen
- kopieren
- verschieben
- zoomen
- drehen (in Gradschritten)
- spiegeln
- dehnen/stauchen/verzerren

Die Elemente des Fensters können aber auch als **SYMBOL** abgespeichert werden. Es ist dann für spätere Zeichnungen in beliebiger Anzahl verfügbar. Die **Funktionstasten** können sogar mit diesen Symbolen belegt werden.

Hervorzuheben sind auch die **Löschfunktionen:** Sie erlauben zum Einen das Entfernen der Elemente in umgekehrter Reihenfolge (das Letzte wird zuerst gelöscht). Außerdem ist es möglich das Element zu löschen, welches dem Fadenkreuz am nähesten liegt. Mit **UNDO** kann der letzte Vorgang rückgängig gemacht werden.

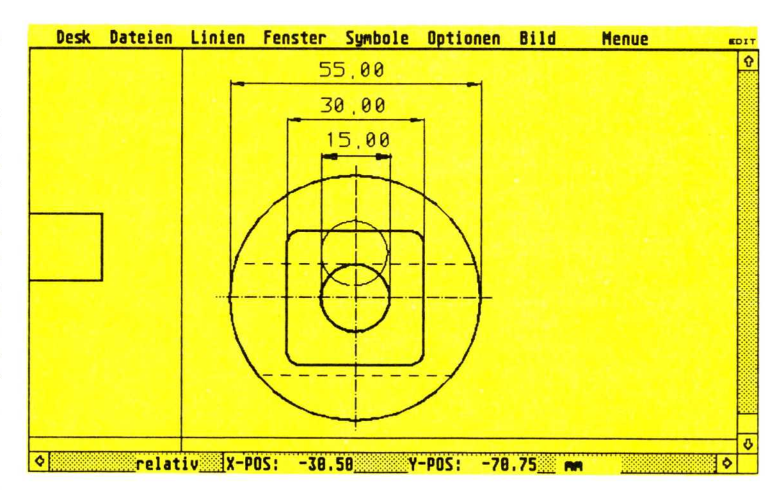


Bild 2: Konstruktionsanwendung

Das Ausgabemedium für ein CAD-Programm ist natürlich ein **Plotter** (20 Plottertreiber werden mitgeliefert). Die meisten Anwender von GFA-DRAFT werden jedoch nur einen **Matrixdrucker** zur Verfügung haben. Deshalb sind auch einige Treiber dafür auf der Diskette (9 und 24 Nadeln). Wessen Drucker oder Plotter nicht angepaßt ist, der kann sich einen eigenen Treiber schreiben. Im Handbuch wird das Vorgehen jedoch mit keinem Wort erwähnt und auch das dokumentierte Beispiel trägt nicht viel zum Verständnis bei. Auf einem NEC P6 ist ein guter, millimetergenauer Ausdruck möglich, auf einem SHINWA CP 80 ist es dagegen noch nicht gelungen.

Da der maßstabsgerechte Ausdruck einer der wichtigsten Punkte ist, sollte man sich vor dem Kauf über die Einsatzmöglichkeit des eigenen Geräts **genauestens informieren.**

Arbeiten mit GFA-DRAFT

GFA-DRAFT arbeitet mit der **Menütechnik**, die als sehr anwenderfreundlich gilt. In diesem Fall ist sie jedoch etwas unglücklich angelegt. Die Funktionen sind teilweise nur schwer zu finden, was auch an der Aufteilung in zwei Menüleisten liegt, zwischen denen hin- und hergeschaltet werden muß.

Die Linienbreite und -art wird über zwei Menüeinträge aufgerufen, obwohl beides meistens zusammenhängt. **Unpraktisch** ist auch, daß die aktuelle Breite und Art nicht angezeigt wird.

Allgemein ist das System etwas schwerfällig, wenn man eine Einstellung verändern will. Ein besserer Menüaufbau oder ein einziges Auswahlmenü wie z. B. bei DEGAS, wären meiner Meinung nach besser gewesen.

Während des Tests traten vereinzelt Fehlfunktionen auf, die jedoch mit der hervorragenden Löschfunktion schnell behoben waren. Bei den FENSTER-Operationen kann es leicht passieren, daß der Inhalt verschwindet. Meistens kommt er nach UNDO wieder zurück, wobei er jedoch innerhalb der Zeichnung auftaucht.

Das **Handbuch** beschreibt manche Funktionen zu kurz und ungenau, so daß man oft erst durch längeres Probieren zum Ziel gelangt. Doch gerade die Hauptanwendergruppe von GFA-DRAFT, die CAD-Einsteiger, werden dadurch unnötig frustriert.

Der **Bildschirmaufbau** ist im allgemeinen sehr schnell, auch beim Verschieben und Zoomen muß nicht lange gewartet werden. Dies ändert sich drastisch, wenn die Linien in ihrer richtigen Dicke dargestellt werden sollen, der Aufbau läßt nun gehörig auf sich warten. Man wird diese Funktion deshalb schnell wieder abstellen, zumal

Linien dabei nur schlecht angepeilt werden können.

Um beim Entwerfen von Grundrissen oder beim Gestalten von Platinenlayouts die Deckungsgleichheit zu gewährleisten, gibt es die Möglichkeit bis zu neun Ebenen übereinander darzustellen. Die Ebenen können auch überlagert ausgedruckt werden.

GFA-DRAFT ist ein durchaus brauchbares CAD-Programm, mit dem maßstabsgetreue Zeichnungen erstellt und ausgegeben werden können. Einige Funktionen sind sehr gut und komfortable, andere dagegen eher gewöhnungsbedürftig und umständlich. Zumindest CAD-Einsteiger, die bei dem günstigen Preis sicherlich die Hauptzielgruppe sind, werden an manchen Stellen ihre Schwierigkeiten haben.

GFA-DRAFT wird mit einem 140-seitigen Handbuch in einem attraktiven Schuber ausgeliefert. Ein Konverter um **MICA**-Dateien zu übernehmen ist auch enthalten. Der Preis ist angesichts der gebotenen Leistung, unter Berücksichtigung der beschriebenen Einschränkungen, und im Vergleich zu anderen CAD-Programmen durchaus angemessen.

(MN)

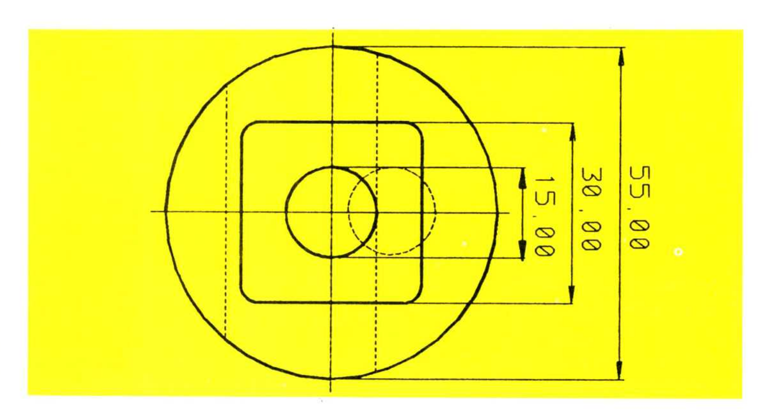
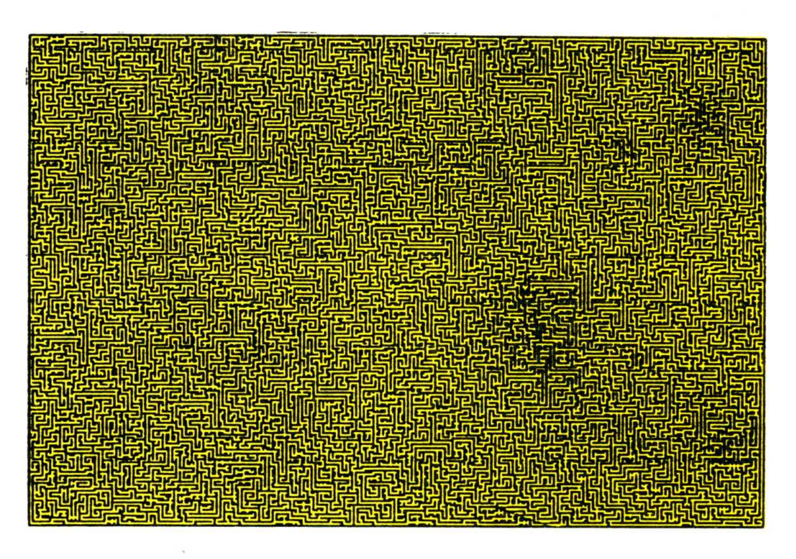


Bild 3: Ausdruck auf dem NEC P6

Der Labyrinthen-Bauer

Wer sich je der Denksportaufgabe gewidmet hat, ein Labyrinth zu konstruieren oder für ein Computerspiel zu programmieren, weiß um die Probleme, die dabei auftreten können. Georg Viehöver erstellte ein Programm, das auf dem Bildschirm ein beliebig großes Labyrinth baut. Darin gibt es zwischen je zwei Punkten stets genau einen Weg.



Das Programm verwendet dabei die Verfahren **Backtracking** und **Rekursion**, die zum Beispiel auch in der künstlichen Intelligenz verwendet werden.

Rekursion bedeutet lediglich, daß sich ein Programm(teil) immer wieder selbst aufruft. Ein bekanntes Beispiel dafür ist die Definition der Fakultät fak(n) (bzw. n!):

fak(n) = 1 ★ 2 ★ 3 ★ ... ★ n.

Das ist die einfache Definition, durch die man fak(n) mit einer einfachen for-Schleife berechnen kann. Die rekursive Definition lautet:

fak(0) = 1
fak(n) = n ★ fak(n−1) für n > 0

Die Definition ist rekursiv, da fak(n) durch fak(n−1) definiert wird, also durch sich selbst. Beispiel:

fak(2) = 2★fak(2−1) (wegen 2>0)
fak(1) = 1★fak(1−1) (wegen 1>0)
fak(0) = 1 (wegen 0=0)

also ist fak(1)=1★1=1, also fak(2)=2★1=2. Wer Lust hat, kann einmal das zugehörige PASCAL-Programm schreiben.

Diese Art zu rechnen ist wesentlich arbeitsaufwendiger als das einfache Durchlaufen einer Schleife. Deshalb bemüht man sich meist, ein **iteratives** Programm (also mit for- und while-Schleifen) zu schreiben. Ein Ergebnis der theoretischen Informatik besagt sogar, daß das immer geht. (Das läuft im Prinzip darauf hinaus, den Stack durch ein vom Programm verwaltetes Feld zu ersetzen und dort z. B. die lokalen Variablen abzuspeichern.) Manchmal

```
program labyrinth;
{entwickelt mit ST-Pascal plus von CCD
 zeichnet Labyrinth beliebiger Größe}

const modulus = 1063;           {Konstanten für Zufallsgenerator}
      factor = 5;
      increment = 7227;
      clr = 0; {Zeichenfarbe}

type tdir = 0..3; {Richtungstyp, 0=hoch , 1=rechts}

var seed, {Zufallszahl}
    xmax,ymax,x,y:integer; {Größe des Feldes, aktueller Bildpunkt}

procedure randomize;       {setzt startzahl auf 'zufälligen' Wert}
begin
   seed:= int(clock() mod modulus)
   {clock liefert Sekunden seit Start}
end;

function random (a,b:integer):integer;
{erzeugt Zufallszahl zwischen a und b}
begin
   seed := (factor * seed + increment) mod modulus;
   random := trunc((seed/modulus)*(b-a+1)+a)
end;


procedure clear_home;         {löscht Bildschirm}
begin write(chr(27),'E') end;

procedure cur_on;             {cursor an}
begin write(chr(27),'e') end;

procedure cur_off;            {cursor aus}
begin write(chr(27),'f') end;

procedure aline(x1,y1,x2,y2:integer);
{zeichnet linie von (x1,y1) nach (x2,y2)}
begin
  line (x1,y1,x2,y2,clr,0,0,0,$ffff,1)
end;
```

```
procedure wait;
{wartet blinkend auf Taste}
begin
   cur_on; repeat until keypress; cur_off
end;


procedure step (var x,y :integer; direction: tdir);
{verändert x,y für einen Schritt nach Richtung}
begin
   case direction of
      0: y:=y-1;
      1: x:=x+1;
      2: y:=y+1;
      3: x:=x-1;
   end {case}
end;

function teststep(x,y:integer; direction:tdir): boolean;
{überprüfe, ob Schritt in Richtung möglich, wenn ja dann true.
 true, wenn nächstes Feld und die umliegenden frei.}

  function testnext(x,y:integer; direction:tdir):boolean;
  {testet, ob Punkt und linker/rechter Nachbar gesehen
   in direction frei sind}
  begin {testnext}
    if get_pixel(x,y)<>clr then {Punkt selbst ist frei}
      case direction of {teste Nachbar}
        0,2: testnext:=((get_pixel(x+1,y)<>clr) and (get_pixel(x-1,y)<>clr));
        1,3: testnext:=((get_pixel(x,y-1)<>clr) and (get_pixel(x,y+1)<>clr))
      end {case}
    else testnext := false
  end; {testnext}

begin {teststep}
   step (x,y,direction); {nächter Punkt}
   if testnext(x,y,direction) then begin {ist frei}
      step(x,y,direction); {übernächster Punkt}
      teststep := testnext(x,y,direction)
   end {then}
   else teststep := false
end; {teststep}

Procedure goline(var x,y:integer; direction: tdir; long:integer);
{geht maximal long Schritte in direction. stoppt, wenn
 Hindernis in Sicht.}
     .
begin
   while(long >0) and teststep(x,y,direction) do begin
        step(x,y,direction); {mache Schritt}
        put_pixel(x,y,clr);  {setze Punkt}
        long := long-1
   end {while}
end;


procedure makestep(x,y:integer);
{die erste Richtung wird zufällig ausgewächselt. Danach werden alle
 richtungen systematisch geprüft. Wenn möglich wird 2..5 Schritte
 weitergegangen.}
var direction: tdir; { Gehrichtung: 0=up, 1=right..3}
    xneu, yneu, {neue Koordinaten}
    count, {zählt getestete Richtung}
    long : integer ; {Anzahl Schritte maximal}

begin
   direction := random(0,3); {erste Gehrichtung bestimmen}
   count := 0; {zählt versuchte Richtung}
   while count <4 do begin {Schleife durchläuft alle 4 Richtungen}
     if teststep(x,y,direction) then begin
        {wenigstens ein Schritt möglich}
        xneu:=x; yneu:=y; {Startpunkt}
        long:=random(2,5); {maximaler Weg}
        goline(xneu,yneu,direction,long); {gehe dahin}
        makestep(xneu,yneu); {Versuche von da weiterzugehen}
     end; {if Step möglich}
     if direction=3 {nächste richtung wählen}
        then direction := 0
        else direction := direction + 1;
     count:=count+1
   end {while}
end;
```

ist es jedoch wesentlich einfacher, ein Problem mit rekursiven Programmen zu lösen. Während eine iterative Problemlösung oft nur sehr unübersichtlich zu programmieren ist, wird bei der rekursiven die Lösungsidee sofort klar.

Backtracking (Rückzug) ist eines der Verfahren, mit dem der Computer eine Lösung durch systematisches Probieren finden kann. Während Menschen (z. B. beim Schach) oft gefühlsmäßig sagen können: "Das ist ein guter Zug", muß der Computer stur und systematisch alle Möglichkeiten durchprobieren und sehen, was passiert. Er findet eine Lösung durch Versuch und Irrtum (trial und error).

Eigentlich ist ein Computer auch nicht klüger als ein Maulwurf, der ein unterirdisches Labyrinth baut: Wenn der an seinem Startplatz ist, überlegt er sich zuerst, in welche Richtung er losgraben und wie weit er geradeaus graben will. Dann versucht (!) er, ob er in diese Richtung graben kann. Wenn es möglich ist, tut er es soweit wie möglich. Anschließend trifft er die gleichen Entscheidungen wie am Startpunkt: Rekursion.

Was passiert, wenn er nicht in die gewünschte Richtung losgehen kann (z. B., weil da eine Hauswand ist, oder weil er dann wieder auf einen alten Gang treffen würde)? Dann probiert er eben die anderen Grabe-Richtungen durch. Nachdem er alles versucht hat und feststellt: "Hier geht's nicht mehr weiter" (das ist der Irrtumsfall), geht er eben zurück zum letzten Eckpunkt (der 'Backtrack'!) und probiert, von dort aus weiterzugraben.

So geht das immer weiter – bis er schließlich wieder an seinem Startplatz ankommt und auch dort nicht weiter kann. Das Labyrinth ist fertig!

Genau das macht auch das PASCAL-Programm, und auch Expertensysteme machen oft nicht viel mehr: Um eine Vermutung zu beweisen, bauen sie eine Kette von Schlußfolgerungen (Versuch) auf, bis sie merken: "Stop, so komme ich nicht ans Ziel" (Irrtum), gehen einen Folgerungsschritt zurück und versuchen es von da aus noch einmal etc., bis sie entweder am Ziel sind oder es nicht mehr weiter geht (Vermutung kann nicht bewiesen werden).

Die zentrale Prozedur im PASCAL-Programm ist MAKESTEP. Sie wählt zufällig die 'Grabe-Richtung' und die Strecke aus (RANDOM), geht sie nach Möglichkeit (GOLINE) und ruft sich selbst auf, um am erreichten Eckpunkt wieder das gleiche zu tun. Danach probiert sie die übrigen Richtungen durch (while-Schleife) und geht schließlich einen Schritt zurück, indem sie zur aufrufenden Prozedur zurückkehrt.

```
begin {hb}
  write(chr(27),'b0',chr(27),'c1');     {Weiß aus Schwarz drucken}
  clear_home;
  writeln('labyrinthmaker');
  write('xmax ymax (<640, <400): '); readln(xmax,ymax);
  x:= xmax div 2; y:= ymax div 2;       {Start in Mitte}
  randomize; cur_off;clear_home;
  aline(0,0,0,ymax);aline(0,0,xmax,0);  {male Box um Feld}
  aline(0,ymax,xmax,ymax);aline(xmax,0,xmax,ymax);
  put_pixel(x,y,clr);                   {Startpunkt setzen}
  makestep(x,y);
  wait;
  write(chr(27),'b1',chr(27),'c0');     {schwarz auf weiß}
end.
```

Die Rolle des Tastsinns beim Maulwurf übernimmt die Funktion TESTSTEP. Sie überprüft, ob man beim nächsten Schritt auf einen anderen Gang stoßen wird oder ob der Weg frei ist.

Damit der Programm-Maulwurf nicht ewig weitergräbt, wird im Hauptprogramm zuerst eine 'Mauer' um das Feld gemalt. Wenn das Programm fertig ist, blinkt der Cursor. Mit <alternate> + <Help> kann man das Labyrinth drucken lassen, mit <Space> wird das Programm beendet.

Drei Bemerkungen zum Schluß:

1. Das Programm braucht bei der Rekursion viel Speicherplatz. Bei einem Feld von 640 ★ 400 kann es Rekursionstiefen bis ca. 12 000 erreichen. Deshalb: Alle wichtigen Sachen vor dem Programmlauf auf Disk abspeichern. Denn wenn der Speicher überläuft, haucht der ST in der Regel sein Leben aus.

2. Ein vergleichbares Programm in Basic auf dem C64 lief etwa 60 mal (!) langsamer. Es war ohne Rekursion programmiert, da Basic nicht richtig rekursiv (mit lokalen Variablen etc.) arbeiten kann.

3. Ein Programm, das einen Weg zwischen zwei Punkten im Labyrinth findet, kann man ebenfalls mit Backtracking programmieren.

Viel Spaß!

(Das Programm wurde entwickelt mit ST-Pascal plus von CCD, Atari 520 ST+, S/W-Monitor, RAM-TOS vom 6.2.86)

Georg Viehöver

Leser & Leser

Leser antworten Lesern

In dieser Rubrik können Sie, liebe Leser, unsere Zeitschrift mitgestalten. Jeder, der Lust und Kenntnis hat, kann seine Frage oder Antwort an uns schicken. Wir werden sie dann veröffentlichen.

Anfrag von Herrn Müller

Sehr geehrter Herr Müller! Da Sie eine Betriebssystemdiskette booten, nehme ich an, das Sie Ihr Philipslaufwerk als Drive B installiert haben. Um Drive B anzusprechen, müßte Ader 12 des Flachbandkabels auf Kontakt 6 des Ataristeckers liegen. Meines Wissens ist in der ST-Computer Nr. 1/86 Ader 14 angegeben worden, was allerdings nicht zutrifft. Zusätzlich sollten Sie die Selektierung des Laufwerks überprüfen. Selektierung Drive A:Jumper 12 gesetzt, Selektierung Drive B:Jumper 13 gesetzt. Ich hoffe Ihnen damit geholfen zu haben. Sollten allerdings Probleme auftreten bin ich gerne bereit Ihnen ausführliche Unterlagen über die Disk Drive zu senden. Viel Erfolg!

Frank Diehl, 4100 Duisburg 1

Phillips X3134 Floppy Laufwerk am ATARI ST

Leseranfrage von Herrn Müller in ST-Computer Nr. 12.

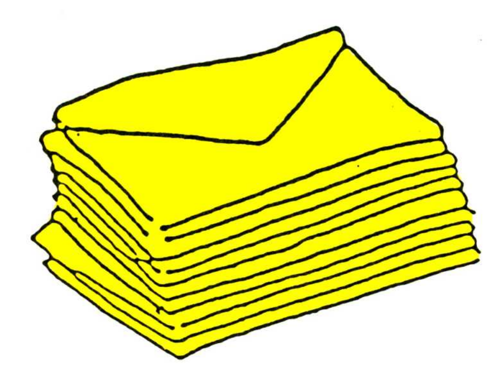
Die Phillipslaufwerke vom Typ X313x haben als Eingangsstufe einen Baustein ULA (Universal Logic Array) 9RB015. Dieser ist laut Angaben im Field Support Manual der Fa. Phillips für diese Drives zwar als voll TTL kompatibel spezifiziert, benötigt aber zum korrekten Arbeiten einen relativ hohen Eingangsstrom. Deshalb muß der Floppy-Datenbus auf diesem Laufwerk mit PULL UP Widerständen abgeschlossen werden. Hieraus ergeben sich aber wegen des Hardware-Designs seitens ATARI zwei Schwierigkeiten, die sich aber lösen lassen.

Als erstes müssen drei Disk-Signale (DRIVE 0 SELECT-Pin10, DRIVE 1 SELECT-Pin12, SIDE SELECT-Pin 32) mit geeigneten Bustreibern, z. B. 7406, versehen werden, weil diese Signale direkt vom Soundchip YM 2149 kommen, der diese Stromlast nicht verträgt. Die Pin-Angaben beziehen sich auf den 34poligen indirekt Steckverbinder.

Als zweites müssen zwei Widerstände im ATARI überbrückt werden, weil sonst durch die höhere Strombelastung der Spannungsabfall an diesen in Reihe geschalteten Widerständen zu groß wird, sodaß die CMOS-Eingangsstufen der 3,5 Zoll Disk-Laufwerke nicht mehr korrekt arbeiten. Die Widerstände (47 Ohm) liegen in der WRITE DATA und in der STEP Leitung und haben die Nummer R100 und R101 (beim ST-260).

In meiner Konfiguration mit zwei 3,5 Zoll NEC FD1035LP als Laufwerke A und B (SEL-0 und SEL-1) und als drittes Laufwerk dieses 5,25 Zoll Phillips (SEL-2) benutze ich ein Widerstandsarray von 330 Ohm im X3134 Laufwerk. Niedriger sollte man nicht gehen, da sich ja auch in den 3,5 Zoll Laufwerken Abschlußwiderstände von jeweils 1 KOhm befinden, die dann zusätzlich auf dem Floppy-Bus parallel geschaltet sind. Abweichend von dem ST-Computer Vorschlag mit dem direkt auf die SF314 Platine gelöteten Flachbandkabel benutze ich eine kleine Platine, die als Eingang den 34 poligen indirekt Stecker des 3,25 Zoll ATARI-Laufwerks hat. Auf dieser Platine habe ich auch den zusätzlichen Bustreiber sowie etwas Logik, die den Betrieb des dritten Laufwerks sowohl manuell über einen Schalter, als auch, mit geeigneter Software, programmgesteuert erlaubt.

Winfried Hübner, 3400 Göttingen



Ich würde mich freuen, wenn Sie mir auf folgende Frage eine Antwort geben könnten: Ist es möglich, eine CANON-Schreibmaschine, die über ein KSR RS232 C Interface verfügt, an einen ATARI Computer (z. B. 260 ST) anzuschließen? Läßt sich dabei der deutsche Zeichensatz ausdrucken (z. B. mit ATARI 1st Word Textverarbeitungsprogramm)?
Claus-Peter Schmid, 8924 Steingaden

Antwort: Bei Anschluß einer Schreibmaschine über eine RS 232 C-Schnittstelle treten normalerweise keine Probleme auf. Ist dem so, so stellt der Ausdruck der deutschen Sonderzeichen mit 1st_Word überhaupt kein Problem dar, höchstens ein Blick ins Druckerhandbuch ist erforderlich. Ob der Anschluß allerdings problemlos funktioniert können wir aus der Ferne nicht beurteilen, deshalb wollen wir diese Frage an die Leser weiterleiten, die eventuell diese Maschine schon erfolgreich angeschlossen haben.

Ich habe mir die TOS-Version vom 6.2.1986 mit 196 480 Bytes besorgt. Leider mußte ich feststellen das diese auf meinem Computer (520 ST+) nicht läuft, d. h. jedesmal wenn ich die TOS Diskette booten will nimmt er mir sie nicht an, obwohl es auf der Diskette drauf ist.
Rolf Füßner, 8900 Augsburg

Antwort: Gerade beim TOS gibt es mehrere Fehlerquellen. Zum einen das TOS-File selbst und zum anderen die Bootspur, die sich auf Track 0 Sector 1 befindet. Die Bootspur ist dringend erforderlich, da ansonsten der Rechner nicht weiß, das er das TOS überhaupt laden soll. Deshalb kann man eine TOS Diskette nur per BACKUP (A - > B, A - > A) kopieren oder nach einem Filecopy die Bootspur mit Hilfe eines Diskmonitors auf die neue Diskette kopieen. Dazu eignen sich alle Monitore (z. B. der altbekannte JOSHUA).

Seit einigen Tagen habe ich ein Problem, bei dem ich einfach nicht alleine weiterkomme: Ich programmiere auf meinem ATARI ST mit ST-PASCAL+ und möchte in selbsterstellten PASCAL-Programmen zusätzliche Zeichensätze verwenden, die auch schon dank EASY-DRAW und DESIGN-SET ST vorhanden sind. Dies müßte mit GDOS2.PRG, ASSIGN.SYS und den Befehlen OPEN WORKSTATION und LOAD FONTS auch möglich sein. Die Frage ist nur wie, mit welchen Parametern und in welcher Reihenfolge diese oder noch zusätzlich notwendige Befehle ausgeführt werden müssen. Ich hoffe, daß Sie mir weiterhelfen können und bedanke mich für Ihre Mühe im voraus.
Uwe Repplinger, 1000 Berlin 42

Antwort: In dieser Kürze können wir Ihnen keine tiefgreifende Auskunft geben. Wir möchten deshalb auf einen im März-Heft erscheinenden Artikel über Zeichensätze verweisen. Für sofortige Auskünfte steht allen rechtmäßigen ST-Pascal Plus Besitzern übrigens die Pascal Hotline zur Verfügung, die Ihnen sicher bei Ihren Problemen helfen kann.

Druckertreiber für Epson LQ-800
Ich weiß, daß das Thema LQ-800 schon öfters in der Leserecke angesprochen worden ist. Mir ist es trotzdem noch nicht gelungen, einen Druckertreiber, der Hardcopies oder EASYDRAW- bzw. DEGAS-Dateien mit 24 Nadeln ausdruckt, aufzutreiben. Robin Höher aus Berlin schreibt zwar von einem kommerziellen Programm von Microtec, aber mir sind 80 DM eigentlich zu teuer für ein Programm, das nur Hardcopies macht. Vielleicht gibt es ja doch schon einen ST-Freak, der so etwas selbst programmiert hat. Da außerdem die Software-Firmen kaum Informationen geben, wie so ein Druckertreiber aussehen müßte, möchte ich anregen, dieses Thema in der ST-Computer zu behandeln. Dabei wäre sicherlich die Überlegung interessant, ob dieser Treiber resident im Speicher verbleibt, und damit evtl. „universell" für alle Programme sein könnte, oder ob er mit dem jeweiligen Programm (EASYDRAW, DEGAS oder TOS) gekoppelt wird.
Thomas Schmitz, 6100 Darmstadt 16

Antwort: Da 24 Nadel Drucker untereinander eine reichliche Kompatibilität aufweisen, kann man dieses Problem mit dem NEC-P6-Treiber, der sich auch auf unserer Public-Domain Diskette 27 befindet, lösen. Dieser Treiber wird auf ein bestimmtes Ausdruck-Format installiert und kann dann jederzeit per 'Alternate+Help' aufgerufen werden.

Assembler-Kurs in ST-C ./12
Ihr Kurs ist sehr übersichtlich und gut verständlich geschrieben, besonders für Anfänger wie ich es noch bin. Trotzdem läuft das abgetippte Programm (garantiert ohne Tippfehler) bei mir nicht. Ich habe den ST 520 auf 1 MB aufgerüstet und arbeite mit dem Assembler „Profimat" von Data-Becker. Wissen Sie Rat? Ich wäre Ihnen sehr dankbar, wenn Sie mir helfen könnten.
Hans Himstedt, 1000 Berlin 47

Antwort: Assembler ist nicht gleich Assembler. Der Standard, der sich auf die Assemblerschreibweise (Syntax) bezieht, ist meist die Motorola Schreibweise. Die Leistungsfähigkeit und die Organisation eines Assemblers bleibt dem Hersteller überlassen, zum Beispiel DATA anstatt .data. Somit ist es Ihre Aufgabe, die nichtgenormten Anweisungen (Pseudobefehle) in die äquivalente, für Ihren Assembler verständliche Sprache zu übersetzen. Dazu sollte im allgemeinen das Handbauch ausreichen, das ja beim Profimat recht umfangreich ist. Dieser Assembler bietet Ihnen aber wesentlich mehr Möglichkeiten Programme zu gestalten, als es in einem Assemblerkurs möglich wäre. Somit muß man allerdings, wegen der Allgemeinheit, auf die Bonbons (z. B. Makros u. ä.) verzichten. Trotzdem wünsche ich Ihnen viel Spaß mit Assembler.

Wir arbeiten derzeit mit einem 1040 ST mit Festplatte und würden gerne wissen, ob es im Zusammenhang mit unserem ATARI die Möglichkeit gibt, Fotosatz herzustellen.

Denkbar wäre doch, daß es ein entsprechendes Programm mit einem Laserbelichter gibt, oder eine ähnliche Einrichtung in Vorbereitung ist.

Wenn Sie uns einen entsprechenden Typ geben können, wären wir Ihnen sehr dankbar und verbleiben für heute mit den besten Grüßen
Anderland Verlagsgesellschaft
8134 Maising

Antwort: In der Sparte News & Infos können Sie näheres über ein solches System lesen, welches die geforderten Fähigkeiten bietet.

Ich habe ein Problem, das ich bereits an die Atari Generalvertretungen der Schweiz und Deutschland, sowie meinem Atari Händler in Zürich erfolglos zur Lösung gab. Das Problem ist folgendes: Ich brauche den Atari ST vor allem zur Textverarbeitung. Mit dem Programm 1st Word bin ich auch sehr zufrieden und es erlaubt eine professionelle Anwendung. Nur mit der Tastatur meines Computers bin ich nicht zufrieden. Sie klemmt manchmal, so das ein flüssiges Schreiben fast unmöglich ist. Ich möchte nun die Leser dieser Zeitschrift fragen ob jemand weiß ob ich eine externe Tastatur kaufen und auch anschließen kann. Vielleicht liegt es auch daran, daß ich einer der ersten 520er gekauft habe und die damals eine schlechtere Tastatur hatten. Weiß jemand wo ich eine neue ST Tastatur für meinen Computer bekommen kann?
Kurt Oertli, CH-8755 Ennenda

Antwort: Externe Tastaturen wurden von mehreren Firmen angekündigt, doch leider blieb es bisher bei der Ankündigung. Sobald eine neue Tastatur lieferbar ist, werden wir sofort darüber berichten.

Mit großem Interesse las ich ihren Artikel über den NEC-P6 Drucker, den ich seit einigen Wochen besitze. Gemäß ihrer Anleitung versuchte ich auch, dem Drucker Proportional spacing unter 1st Word beizubringen, mußte jedoch feststellen, daß diese Lösung nur bedingt lauffähig ist.

Zwar funktioniert der Blocksatz tadellos, doch gerade das ist das Problem. Bei Absatzenden, wie sie eines weiter oben im Text erblicken können, wäre das Resultat mit der im Heft abgedruckten Anpassung unbrauchbar. Die letzte Zeile, die ja bewußt nicht auf Blocksatz getrimmt werden soll, wird nun auf die volle Zeichenbreite gedehnt. Ich brauche aber für meine Arbeit vernünftig gestaltete Absätze.
Peter A. Kemény, 6000 Frankfurt 1

Antwort: Das Manko der Ignoranz des Absatzendes ließ sich der Drukker leider nicht abgewöhnen. Die vorgeschlagene Steuerzeile hatten wir allerdings nicht als Anpassung, sondern nur als kleinen Tip vorgesehen und erheben keine Garantie auf Vollständigkeit. Falls ein Leser eine Lösung gefunden hat, so sind wir gerne bereit diese zu veröffentlichen.





WIZARD ROYAL

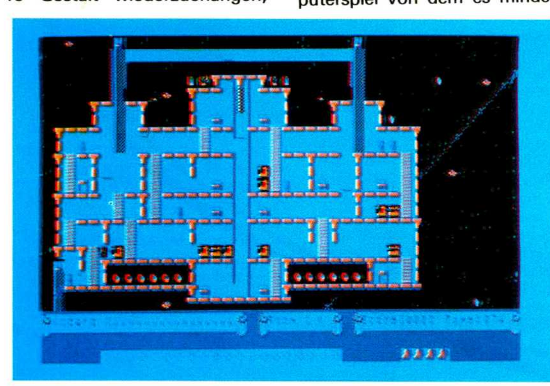
Ein Spielprogramm für lange Winterabende ist WIZARD ROYAL. Kurz gesagt ist es ein Grafik-Adventure, bei dem jederzeit neue Spielszenen mit einem Editor erstellt werden können. Die Spielidee ist dabei folgende: in fernen, vergangenen Zeiten, wogte der Kampf zwischen Gut und Böse auf der Erde. Unglücklicherweise unterlag das Gute und die 'Gilde der Zauberer' wurde in Eulen verwandelt. Um ihre Gestalt wiederzuerlangen, müssen die Eulen nun durch Zeiten und Kontinente reisen und viele Abenteuer und Gefahren bestehen. Sie müssen versteckte Zauberkisten finden und in die 'Höhle der Ewigkeit' bringen, erst dann kann der Fluch aufgehoben werden.

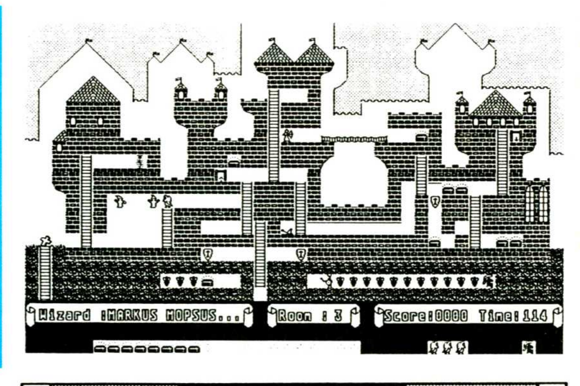
Die Szenerien dazu sind sehr umfangreich: 20 Levels mit je bis zu 5 Spielszenen können durchlaufen werden, wobei der Wechsel zwischen den Szenen über 'Zeit-Tore' erfolgt. Nützlich sind dabei: Zeitkrüge, Schlüssel, welche die Höhlen der Ewigkeit öffnen, Glückssymbole und Zauberhelme, die vor Gefahren schützen. Jede Szene stellt andere Anforderungen an den Spieler; so wird einmal mehr Geschicklichkeit und ein andermal mehr Logik und Auffassungsgabe verlangt.

Wizard Royal hat eine gewisse Ähnlichkeit mit 'LODE RUNNER', einem erfolgreichen Computerspiel von dem es mindestens drei Teile gibt. Wizard Royal ist noch abwechslungsreicher gestaltet, erlaubt mehr Handlungsspielraum und die Funktionen der Editoren sind erheblich umfangreicher. Dies zeigt sich auch in dem Platz, der deren Beschreibung im Anleitungsheft gewidmet wird. Rund dreiviertel der Seiten beschäftigen sich mit dem Symbol- und Leveleditor, und entsprechend umfangreich sind auch die dort festlegbaren Werte. Die Möglichkeiten sind demnach sehr vielfältig und somit wird das Spiel den Fan lange beschäftigen.

Beinahe hätte ich jetzt vergessen, den Sound des Spiels zu erwähnen, der jedoch nicht unbeachtet bleiben darf. Schon gleich im Eingangsbild murmelt ein Zauberer eine fremdländisch klingende Zauberformel. Und im Spiel sind noch viele weitere Toneffekte eingearbeitet (z. B. eine knarrende Tür beim Betreten eines 'Zeit-Tores'), die, da sie digital aufgenommen wurden, sehr gut klingen.

Die Grafik ist sehr abwechslungsreich, allerdings ist der Hintergrund öfter etwas karg. Bei der Farbversion (wird zusammen mit der monochromen geliefert) sind die Farbtöne zum Teil sehr kontrastarm, wodurch gerade auf einem Farbfernseher Einzelheiten nur schwer zu erkennen sind.



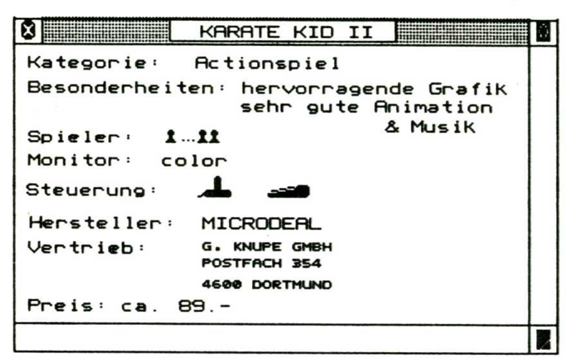




WIZARD ROYAL

Kategorie: Grafikadventure
Besonderheiten: Levelgenerator/Editor
digitaler Sound
Spieler: 1...11111
Monitor: monochrome & color
Steuerung:
Hersteller: MEGASOFT, Neustadt
Vertrieb: Boston-Computer GmbH
Anzinger Str. 1
8000 München 80
Preis: ca. 139.-

KARATE KID Part II



KARATE KID II

Kategorie: Actionspiel

Besonderheiten: hervorragende Grafik
sehr gute Animation
& Musik

Spieler: 1...2

Monitor: color

Steuerung:

Hersteller: MICRODEAL

Vertrieb: G. KNUPE GMBH
POSTFACH 354
4600 DORTMUND

Preis: ca. 89.-



Nach ST-Karate, das in der letzten ST-Computer vorgestellt wurde, ist nun KARATE KID II auf dem Markt erschienen. Die Spielhandlung bezieht sich wie auch der Titel auf den gleichnamigen Film.

Der junge Daniel LaRusso kämpft gegen verschiedene, immer stärker werdende Gegner, bis er am Ende des Spiels auf den gefürchteten Chozen trifft. Dieser, alles entscheidende Kampf findet auf der Festung des Königs Shohashi statt. Allerdings hat es mit dieser Szene etwas Besonderes auf sich, denn während des Kampfes erscheint eine geheimnisvolle Trommel, deren Zweck man herausfinden muß, um den Kampf zu überstehen.



Die Joysticksteuerung ist fast die gleiche, wie bei ST-Karate, allerdings sind die Abläufe schneller und die Figuren wesentlich besser animiert.

Die Akteure gehen so lange zu Boden, bis ihr Kraftbalken aufgebraucht ist. Daniel, der immer in weiß gekleidet ist, hockt nach manchen Schlägen eine Weile auf der Erde und und kratzt sich am Kopf, bevor er weiterkämpft.

In dem Moment, in dem der Energiebalken aufgebraucht ist, erhebt sich der Kämpfer und kurz darauf liegt er wieder langgestreckt auf dem Boden. Wie schon gesagt, die Bewegungen und Handlungen der beiden Karatekas sind wirklich ausgezeichnet. Das Gleiche gilt auch für die Hintergrundgrafik. Sie übertrifft sogar die wirklich nicht schlechte des ST-Karate-Spiels.

Die meisten Kämpfe finden innerhalb eines Raumes statt, dessen Wandgemälde und Ausblick jedoch immer verschieden sind. Die Szene besticht durch die Vielzahl von Details, die dargestellt werden und den asiatischen Charakter der Handlung unterstützen. Sehr schön ist auch die letzte Sequenz, die auf einem Dach der Festung, im bläulichen Dämmerlicht des Mondes, stattfindet.

Zwischen den beschriebenen Szenen gibt es noch Zwischenspiele, bei denen der Bezug zum Film deutlicher wird, weil hier dessen Hauptcharaktere eine größere Rolle spielen. In der ersten Handlung muß Miyagi mit chinesischen Eßstäbchen eine Fliege fangen. Diese Übung ist nicht leicht und erfordert viel Konzentration.

Die zweite Szene zeigt Daniel vor einem Stapel Eisplatten. Durch schnelle Joystickbewegungen holt er Schwung und schlägt dann auf die Platten. Je mehr Schichten durchbrechen, desto mehr Punkt erreicht man.

In beiden Szenen werden digitalisierte Bilder verwendet. Miyagi ist dadurch sehr gut zu erkennen, allerdings ist diese Konzentrationsübung auf die Dauer so lästig wie die Fliege, die man fangen soll. Dies fällt zudem dann besonders auf, wenn das Spiel zu zweit gespielt wird und jeder die Zeit durchstehen muß. Dies ist allerdings der einzigste Kritikpunkt an diesem Programm.

Daniel ist nicht so gut zu erkennen, dafür ist die Eisbrecherszene aber wesentlich besser und nicht so langatmig.

Auch musikalisch hat das Spiel etwas zu bieten. Schon nach einer kurzen Ladezeit beginnt die Musik zu spielen, wobei es sich um das Titellied des Films 'The Glory of Love' handelt. Ihre Umsetzung auf den Computer ist gut gelungen, und auch die anderen Geräuscheffekte kommen gut an.

KARATE KID II bietet eine wirklich hervorragende Grafik und eine ebenso gute Animation der Figuren. Wem also ST-Karate schon gut gefallen hat, der wird von KARATE KID begeistert sein.



PATIENCE


PATIENCE

Kategorie: Denkspiel

Besonderheiten: sehr gute Grafik
Hilfsfunktionen

Spieler: 1

Monitor: monochrome

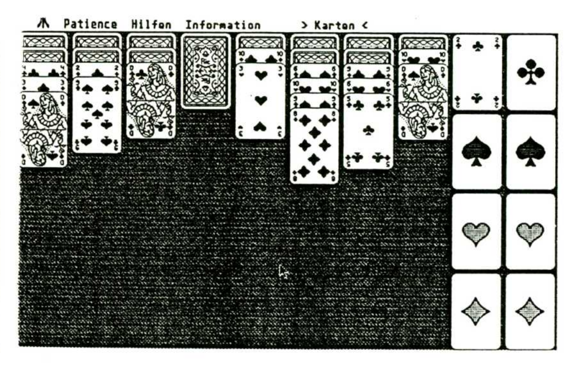
Steuerung:

Hersteller: Volker Weidner

Vertrieb:

PUBLIC DOMAIN

Preis:

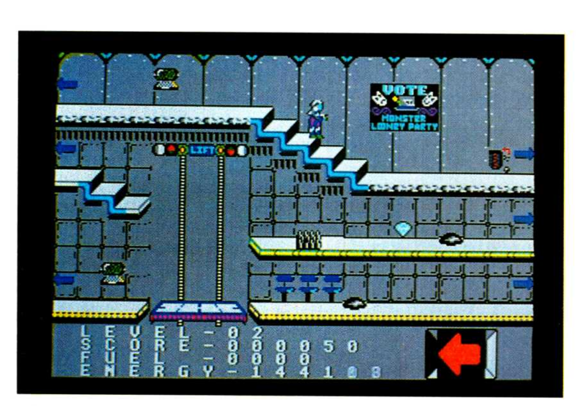


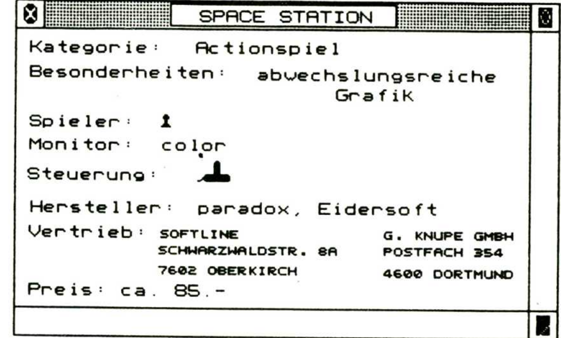
PATIENCE ist eine Variante der bekannten Kartenlegespiele, die sich großer Beliebtheit erfreuen. Um es kurz zu beschreiben: Ausgehend von einem gemischten Kartenstapel werden die Karten erst ausgelegt und dann nach bestimmten Regeln umgruppiert. Dabei wird versucht, alle Karten in eine aufsteigende Reihenfolge zu bringen. Wenn dies im Rahmen der Vorschriften gelingt, ist die Patience 'aufgegangen'. Dazu ist jedoch einige Übung, eine gute Kombinationsgabe - und natürlich auch Geduld (patience (franz.): Geduld) notwendig.

Zu Beginn des Spiels wird eine Anzahl von Karten auf sechs Reihen verteilt. Nun kann man auch die gute grafische Gestaltung des Spiel erkennen. Besonders die Bilderkarten sind mit viel Sorgfalt erstellt worden. Das Programm übernimmt von nun an die komplette Kontrolle über das Spiel. Das bedeutet, daß jede Handlung des Spielers auf ihre Richtigkeit überprüft wird. Außerdem können Zugvorschläge angefordert werden und eine anwählbare Automatik übernimmt das Umlegen aller möglichen Karten auf die Zielstapel. Wer in eine hoffnungslose Situation geraten ist, in der kein Zug mehr möglich ist, der kann versuchen sich über einige Zugrücknahmen zu retten. Sollte dies Fehlschlagen, gibt es noch die Möglichkeit, die Kartenstapel beliebig zu verändern. Nach einem solchen Eingriff wird jedoch keine Gratulation mehr ausgegeben, wenn die Patience aufgeht.

Wer jetzt noch nichts verstanden hat, dem wird die Funktion 'Anleitung' weiterhelfen und die Information, daß dieses Spiel Public Domain (frei kopierbar!) ist, sollte jeden Interessierten veranlassen, sich dieses Programm sofort zu bestellen. Das Programm bietet höchsten Bedienungskomfort und ist optisch so reizvoll, daß man dem Autor dafür nur danken kann. Viele werden jetzt Stunde um Stunde die Herausforderung der Patience auf sich nehmen.

SPACE STATION




SPACE STATION

Kategorie: Actionspiel

Besonderheiten: abwechslungsreiche
Grafik

Spieler: 1

Monitor: color

Steuerung:

Hersteller: paradox, Eidersoft

Vertrieb: SOFTLINE
SCHWARZWALDSTR. 8A
7602 OBERKIRCH

G. KNUPE GMBH
POSTFACH 354
4600 DORTMUND

Preis: ca. 85.-

Sie sind Callum MacGregor, Chef-Inspektor der Vereinigung der Raumstaitonen, und Ihr Auftrag ist, die rätselhaften Vorkommnisse auf ALPHA 3 zu untersuchen.

Als Sie dort ankommen, wird Ihnen die Situation klarer. Die Raumstation wurde von außerirdischen Wesen angegriffen und erobert. Keiner der Bewohner hat überlebt. Das Einzigste, was Sie noch tun können ist, den Selbstzerstörungsmechanismus der Station in Gang zu setzen. Dafür muß jedoch die Sicherheitsdiskette gefunden werden, die irgendwo herumliegt. Auf der Suche sollten auch alle Lithium-Kristalle mitgenommen werden, da diese als Energiequelle für die Raumfähre Ihren Rückflug erst ermöglichen, oder wollen Sie etwa für immer dort herumirren?

Herumgeirrt wird bei diesem Spiel eine ganze Menge. Die Raumstation ist ein einziges Labyrinth aus Gängen und Ebenen, eine riesige Anlage mit mehr als 100 Räumen. Als ob dies alleine nicht schon ausreichen würde, gibt es auch noch Stolpersteine und Teleporter. Sie katapultieren im günstigsten Fall eine Ebene höher oder aber an eine völlig andere Stelle in der Station.

Überall lauern Gefahren und Fallen, die zum Teil nur durch geschicktes Handeln und Reagieren zu bewältigen sind.

Die Szenerie von SPACE STATION ist farbenfroh und sehr abwechslungsreich. Die Musik, die während der ganzen Zeit die Handlung untermalt, ist zwar nicht schlecht, doch auf die Dauer etwas ermüdend. Obwohl das Spiel an sich etwas einfach ist, kann ihm ein guter Unterhaltungswert zugeschrieben werden.

...UND PLÖTZLICH HAT IHR ATARI ST DAS LAUFWERK DER PROFIS

z. B. vortex MA1-D

Ein 3.5" Doppellaufwerk der jüngsten Generation mit 1,4 MB formatierter Speicherkapazität. **Kompakt:** 290 (L) x 105 (B) x 64 (H) mm. **Komplett:** Integriertes Netzteil (25 W, 220 V, 50 Hz). **Kompatibel:** Atari-gleicher Floppy-Stecker; Atari-gleiche Lackierung. **Kostengünstig:** Nur 998,– DM*. Den vortex MA1 gibt es auch als Single-Laufwerk zum späteren Aufrüsten. Außerdem im vortex-Atari-Tuning-Programm: 5.25"-Einzel- und Doppellaufwerke. 3.5"- plus 5.25"-Laufwerk („Gemischtes Doppel").

vortex verwendet NEC- bzw. BASF-Laufwerke.

*empfohlener Verkaufspreis.



I·N·F·O-S·C·H·E·C·K

Bitte senden Sie mir weitere Informationen über Ihre „Atari"-Laufwerke und einen Händlernachweis.

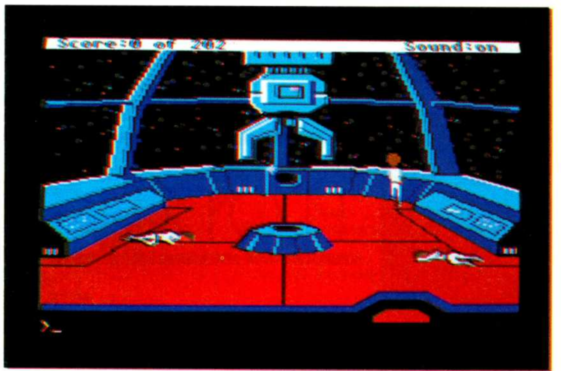
ST

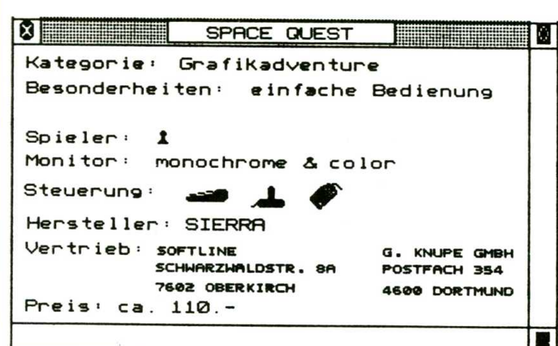
vortex Computersysteme GmbH · Falterstraße 51-53 · 7101 Flein



...UND PLÖTZLICH HABEN SIE EINEN PROFI-COMPUTER

SPACE QUEST





SPACE QUEST

Kategorie: Grafikadventure
Besonderheiten: einfache Bedienung

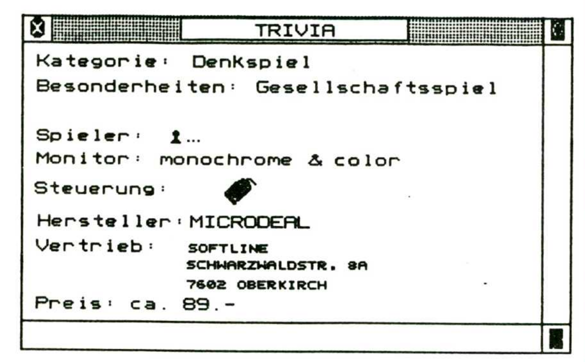
Spieler: 1
Monitor: monochrome & color
Steuerung:
Hersteller: SIERRA
Vertrieb: SOFTLINE, SCHWARZWALDSTR. 8A, 7602 OBERKIRCH — G. KNUPE GMBH, POSTFACH 354, 4600 DORTMUND
Preis: ca. 110.-

Weit draußen im All liegt das Planetensystem von EARNON, dessen Sonne seit langem im Sterben liegt. Das Ende scheint bald bevorzustehen, doch im Raum-Laboratorium ARCADA haben Wissenschaftler gerade ihre Versuche für einen STERNENGENERATOR abgeschlossen. Mit dessen ungeheuren Energie soll es möglich sein, einen verlöschenden Planeten in eine neue Sonne zu verwandeln. Die ARCADA befindet sich mit dieser guten Nachricht gerade auf dem Rückflug, als sie von außerirdischen Wesen, den SARIENS, angegriffen wird, die die Kraft des Sternen-Generators zur Unterdrückung des Universums einsetzen wollen.

Die SARIENS übernehmen die Kontrolle über das Raumschiff, und alle Hoffnungen für den Planeten EARNON scheinen dahin zu sein. Oder gibt es etwa Rettung vom Planeten KERONA, oder von der superintelligenten Rasse, die unter der Erde wohnt? Dies ist die Situation, in der Sie sich befinden und alles hängt von Ihnen ab - also los geht's.

Ganz im Stil von KINGS QUEST ist auch dieses neue Grafikadventure von SIERRA. Die Grafik ist bei diesem Adventure interessant und reizvoll, zumindest wenn man sie auf dem Farbbildschirm bewundern kann. In schwarzweiß sieht alles nicht mehr so rosig aus, die Grafi ist hier deutlich schlechter. Die Bedienung kann über Tastatur, Maus oder Joystick erfolgen und ist sehr einfach. Als Eingabe werden nur kurze Befehle erwartet (explore, pick up, use, talk). Somit kann man sich ganz der Handlung hingeben und die Rettung für EARNON herbeiführen. Die Fortsetzung der Saga ist bereits für nächstes Jahr geplant.

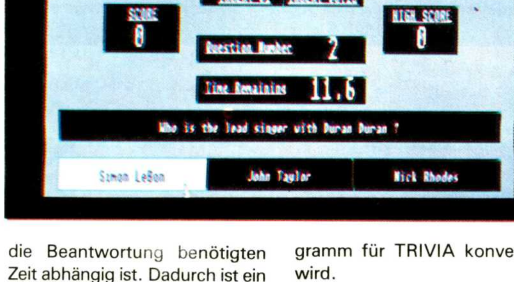
TRIVIA CHALLENGE I



TRIVIA

Kategorie: Denkspiel
Besonderheiten: Gesellschaftsspiel

Spieler: 1...
Monitor: monochrome & color
Steuerung:
Hersteller: MICRODEAL
Vertrieb: SOFTLINE, SCHWARZWALDSTR. 8A, 7602 OBERKIRCH
Preis: ca. 89.-



Vermutlich kommt Ihnen der Titel bekannt vor, oder gehören Sie etwa nicht zu den begeisterten Spielern von Trivial Persuit, einem Spiel, das gegen Ende des letzten Jahres auch auf den deutschen Markt kam.

Bei TRIVIA werden aus den Wissensgebieten Sport, Kunst, Popmusik, Wissenschaft und Allgemeinwissen Fragen gestellt und dazu jeweils drei Antworten angeboten. Sie müssen dann nur noch die Richtige auswählen. Schon haben Sie gewonnen und dürfen weiterspielen. Damit das Ganze nicht zu einfach wird, ist ein Zeitlimit vorgegeben, innerhalb dem Sie sich entscheiden sollten, sonst wird die Frage als falsch gewertet. Das Spiel setzt sich so lange fort, bis Sie zwei Fragen falsch beantwortet haben. Das kann jedoch ganz schnell gehen, denn so einfach sind die Fragen nicht.

Für jede richtige Antwort gibt es Punkte, deren Höhe von der, für die Beantwortung benötigten Zeit abhängig ist. Dadurch ist ein fairer Wettstreit zwischen verschiedenen Kontrahenten möglich, wobei jeder auf hohe Punktzahlen spekulieren wird.

Der Fragenkatalog von TRIVIA ist durch eigene Kreationen erweiterbar, indem ein neues Datenfile erstellt wird. Dazu wird mit einem Textverarbeitungsprogramm ein Dokument mit Fragen und Antworten erstellt, daß dann von einem separaten Programm für TRIVIA konvertiert wird.

Von der optischen Gestaltung her hat TRIVIA nicht viel zu bieten, und auch sonst sind keine Besonderheiten vorhanden. Rund 4000 Fragen hält TRIVIA bereit, wobei ein Teil sehr speziell und auf ein amerikanisches Publikum abgestimmt ist. Als kleiner Trainer auf dem Gebiet der 'Allgemeinbildung' und vor allem für gesellige Runden ist TRIVIA wie auch seine äquivalente allemal gut geeignet.

...UND

PLÖTZLICH

ERREICHEN

SIE

MIT IHREM

ATARI ST

UNGEAHNTE

DIMEN-

SIONEN



Mit der vortex HD20-Station.

Ein Hard-Disk-Laufwerk („Winchester") mit 21 MB formatierter Speicherkapazität. **Das sind rund 10.000 vollgeschriebene DIN A 4-Seiten!** Dazu der blitzschnelle Zugriff: 85 ms! Die vortex-HD20 ist komplett und kompakt: Netzteil (32 W, 220 V), „Winchester"-Laufwerk und HOST-Adapter sind in einem Atari-weißen Gehäuse (L 300 mm x B 110 mm x H 64 mm) untergebracht. Systemdiskette und ausführliches deutsches Handbuch werden mitgeliefert. **Machen Sie aus Ihrem Atari ST einen echten Profi-Computer. Nutzen Sie unser Test-Angebot.**



BESTELL-COUPON

☐ Senden Sie mir die vortex-HD20-Station versandkostenfrei per Nachnahme. Ich kann sie ohne Angabe von Gründen innerhalb 10 Tagen zurücksenden und erhalte den Kaufpreis in Höhe von DM 1.798,- umgehend erstattet.

☐ Ich möchte nähere Informationen über die vortex-HD20-Station.

ST

Name ______________________

Anschrift ______________________

Unterschrift ______________________

Datum ______________________

vortex Computersysteme GmbH · Falterstraße 51-53 · 7101 Flein

vortex
COMPUTERSYSTEME

...UND PLÖTZLICH HABEN SIE EINEN PROFI-COMPUTER

The Pawn

– Das Adventure des Jahres 1986



...25.10.1986 3^{00} Uhr nachts: Wieder einmal sitze ich an meinem Rechner und versuche, mich ein wenig zu entspannen. Was gibt es besseres an einem Computer – außer natürlich kreativ zu sein und zu programmieren – als sich einem kleinen Spielchen zu widmen? Viele behaupten, sie könnten sich bei so manchen Ballerspielen, deren Ausführung wie zum Beispiel bei XTRON inzwischen schon fast als hervorragend zu bezeichnen ist, so richtig abreagieren. Sei's drum, auch das braucht der Mensch. Ich allerdings bevorzuge da eine eher andere Gattung der Computerspiele: strategische Spiele wie zum Beispiel KAISER oder als die zweite Möglichkeit: Adventures!

Adventures sind an und für sich nichts anderes als Rollenspiele: Man wird in eine bestimmte Situation gesteckt und soll dann sehen, wie man da nun wieder herauskommt – oft muß man dabei noch eine Prinzessin retten oder vielleicht auch einen Drachen besiegen.

Die ersten Adventures wurden noch in reiner Textform verfaßt, so daß der Computer – manchmal in der Rolle eines Wegbegleiters – dem Spieler mitteilte, wo man sich befand und ihm vielleicht ab und zu einen Tip gab, um des Rätsels Lösung ein wenig näher zu kommen. Einen Höhepunkt dieser Art von Adventures stellt sicherlich die Firma INFOCOM dar, die mit ZORK I, II und III berühmt wurde. Die meisten der INFOCOM-Adventures gibt es auch auf dem ST: Einer der besten Adventures ist HITCHHIKER'S GUIDE TO THE UNIVERSE, das genauso wie die Vorlage (das Buch), den Leser bzw. Spieler regelrecht auf den Arm nimmt.

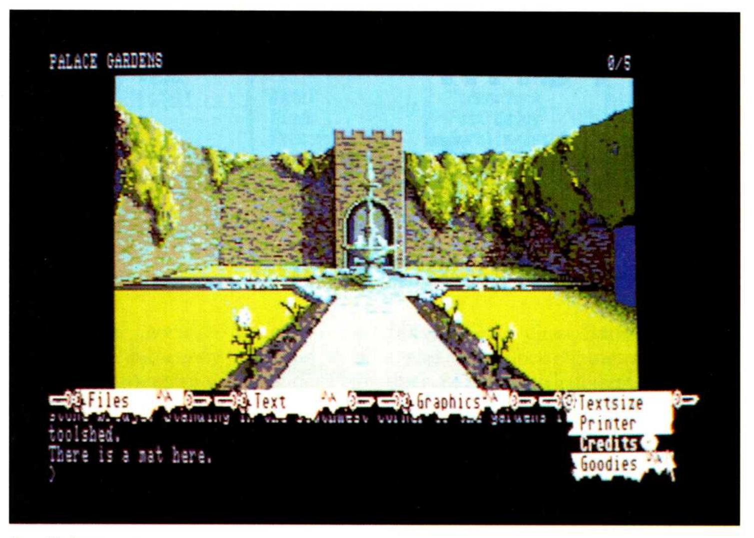
Vor einigen Jahren dann kamen die ersten Grafikadventures auf den Markt, die zu den meisten Orten des Geschehnisses ein Bild lieferten. Leider waren diese Grafiken noch sehr schlecht, was mit daran lag, daß Speicherplatz und Grafikauflösung der Home-Computer noch sehr zu wünschen übrig ließen. Der Effekt war, daß man wieder auf die alte Form der Adventures zurückgriff, da die Grafik eher störte, als daß sie zum Weiterspielen anregte. Bald aber gaben sich auch auf diesem Markt die Spiele-Schreiber noch mehr Mühe: Adventures wie zum Beispiel THE DALLAS QUEST entstanden, die man bis dahin kaum für möglich gehalten hatte.

Nun hat die Firma RAINBIRD vor einigen Monaten das Adventure THE PAWN auf den Markt gebracht, das so einiges auf dem Markt schlägt, was es zu kaufen gibt. Es gehört zu der Kategorie der Grafikadventures, allerdings ist diese Einstufung wohl eher untertrieben, denn die Grafiken die THE PAWN Liefert, sind als sehr gut zu bezeichnen. Gerade dies animiert den Spieler immer wieder, tiefer in das Geheimnis des Adventures einzudringen. Die ganze Aufmachung des Spiels muß von Grund auf durchdacht worden sein, denn das, was man für sein Geld bekommt, ist wirklich enorm. Allein die Verpackung macht schon einen sehr guten Eindruck, allerdings kann dieser Eindruck oftmals täuschen. Eine feste Verpackung, die grafisch sehr schön gestaltet worden ist: Die Zeichnung auf der Verpackung, die als Titelbild im

Programm zu sehen ist, wird sogar als kleines Poster mitgeliefert. Das Beste allerdings ist ein kleines Büchlein, das keiner, der ein wenig Englisch versteht und gerne Adventures (besonders dieses) spielt, missen sollte. Es enthält außer ein paar Tips (dazu später mehr) eine famos erzählte Geschichte aus dem Land KEROVNIA, in dem sich THE PAWN abspielt. Ich möchte hier keine Inhaltsangabe machen; ich kann aber jedem empfehlen, der das Adventure besitzt oder kauft, sich die Geschichte einmal zu Gemüte zu führen; es lohnt sich.

Nun zu dem Adventure selbst: In THE PAWN geht es darum, daß man eines Tages aus einer Bewußtlosigkeit aufwacht und sich in einem fremden Land befindet, das eine Mischung aus Gegenwart und Mittelalter darstellt: So telefoniert in der Geschichte auch mal eine Königin mit der anderen. Das Ziel des Adventures ist im Prinzip ganz einfach: Man versucht, aus dem Land Kerovnia zu entkommen. Um dies zu erreichen, muß man sich wie in anderen Adventures, über Text mit dem Computer verständigen. Auch hier streicht THE PAWN ein dickes Lob ein. Der Programmteil, der die Sätze auswertet und darauf entsprechend reagiert, Parser genannt, enthält derart viele Vokabeln, daß er auch Sätze wie (Englisch-Freaks aufgepaßt) PLANT POT PLANT INTO PLANT POT versteht!

Sicherlich wird man bei jedem Programm etwas zu nörgeln oder zu beanstanden wissen, so hätte man meines Erachtens etwas mehr Bilder dazuliefern



Im Palastgarten

können und außerdem ist das Adventure äußerst schwer lösbar. Man wird zwar nicht alleingelassen, denn in dem kleinen Büchlein befinden sich als Zahlen verschlüsselte Tips, allerdings empfand ich diese Art der Codierung als sehr mühsam, wo es doch inzwischen andere bessere Methoden gibt. Trotzdem ist auch nach Anforderung von zusätzlichen Tips bei RAINBIRD (Sie reagierten sehr schnell, DANKE) ohne große Fantasie schier unmöglich, weiterzukommen.

Alles in allem ist THE PAWN ein rundum gut gemachtes Produkt, das besonders durch seine gute Grafik und den guten Parser hervorsticht. Die ganze Aufmachung (Verpackung, Poster, Büchlein etc.) sollte hier noch mal erwähnt werden. RAINBIRD hat weitere Adventures mit den Namen JEWELS OF DARKNESS und SILICON DREAMS angekündigt. Wollen wir hofen, daß diese Produkte genauso gut werden wie THE PAWN.

Ich jedenfalls werde mir diese Wartezeit damit vertreiben, daß ich versuche, an den Felsbrocken vorbeizukommen, um dort endlich das zu finden, was der Guru von mir möchte, damit er mir wiederum endlich einen Tip gibt, wie ich weiterkomme... oder kann mir vielleicht einer der Leser weiterhelfen?

(Stefan Höhn)

ST-Term Deluxe-Term

Eine Gegenüberstellung



In der Redaktion stapeln sich inzwischen Programme, die darauf warten getestet zu werden. Auch in der Sparte DFÜ- bzw. Terminalprogramme weitet sich das Angebot immer mehr aus. Die Spannweite der angebotenen Terminalprogramme für den ST reicht vom einfachen Low-Cost-Programm bis zur „Luxussoftware".

In dieser Ausgabe werden zwei Terminalprogramme untersucht, die beide weniger als einhundert Mark kosten. Die Kandidaten heißen: „ST-Term" (Franzis Software) und „Deluxe-Term" (HIB). Wie sich im Test sehr schnell herausstellte, sind beide Programme völlig verschieden ausgelegt.

Minimalprogramm?

ST-Term wurde ausschließlich in Assembler programmiert und verzichtet auf alles was man üblicherweise als GEM-Benutzeroberfläche bezeichnet. Die Maus hat keine Funktion. Eine Menüleiste gibt es ebensowenig wie Pulldown-Menüs. Ergebnis dieses Konzeptes ist ein Programm mit weniger als 6000 Bytes Länge, das angeblich so schnell ist, daß man gleichzeitig eine Datei empfangen und eine Datei senden kann. Alle Einstellungen werden durch Funktionstasten vorgenommen. Die obersten zwei Zeilen dienen der Kommunikation zwischen Programm und Benutzer. Die „Help"-Taste bewirkt die Anzeige der Bedeutung aller Funktionstasten (Bild 1). Wie Sie sehen können, sind in „ST-Term" nur die in einem DFÜ-Programm unbedingt notwendigen Features eingebaut. Einziger Luxus ist „F10". Mittels dieser Taste kann eine vorher festgelegte Zeichenfolge abgeschickt werden.

„F1"
soll Ein- und Ausschalten des Modems bewirken. Als Nebeneffekt werden die „In Use"-Leuchtdioden beider Floppy-Disk-Laufwerke manchmal mit eingeschaltet.

„F2" und „F3"
ermöglichen es dem Benutzer Dateien von der Diskette abzusenden, bzw. die empfangenen Daten in eine Datei zu schreiben. Nach Drücken von „F3" werden alle ankommenden Daten auf Diskette gesichert. Bei der „Unmenge" von brachliegendem Ram hätte es sich unbedingt angeboten; ankommende Daten im Speicher zu puffern. Man könnte z. B. beim Stöbern in einer Mailbox den Puffer auf Diskette sichern, wenn man entdeckt, daß etwas interessantes ankam.

„F4"
Alle üblichen Baudraten bis zu 9600 können eingestellt werden.

„F5"
Diese Funktion dient dem Umschalten der V24-Parameter Datenbits, Parität und Stopbits. In dieser Reihenfolge (Beispiel „7n1") stehen einige Einstellungen zur Auswahl.

„F6"
ermöglicht die Kommunikation auch mit anderen Rechnern als mit ST's und IBM's. Deutsche Umlaute werden beim Senden und Empfangen entsprechend umgewandelt.

„F7"
schaltet „XON – XOFF" an und aus.

„F8"

„F9"
schaltet das lokale Echo an und aus.

F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Modem	Upload	Download	Baud	Format	Code	Protokoll	Modus	Echo	Paßwort

Bild 1: Das Menü von ST-Term

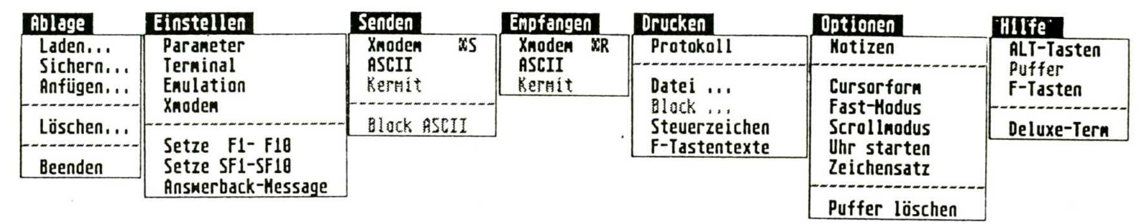


Bild 2: Die Drop-Down Menüs von Deluxe-Term

„F10"
Siehe oben.

Einige Punkte verdienen besonders kritische Bemerkungen: Warum wird in den obersten zwei Zeilen nicht grundsätzlich das Menü angezeigt? Statt in der zweiten Zeile anzuzeigen, daß die Help-Taste das Menü auf den Bildschirm bringt, könnte doch wohl auch gleich das komplette Menü angezeigt werden!

Die Fehlfunktion beim Modemanschalten ist zwar nur ein Schönheitsfehler, aber sowas wäre allenfalls bei Public-Domain-Software entschuldbar.

Das Fehlen eines Standardprotokolls zur Fehlererkennung macht Übertragungen von Programmen zum Glücksspiel.

Luxusprogramm!!

Alle Minimalfunktionen (siehe oben) kann Deluxe-Term auch (gleichzeitiges Senden und Empfangen von Dateien wurde nicht erprobt – wozu auch).

Während ST-Term sich von Terminalsoftware für „normale" Computer weder in der Bedienung noch auf dem Bildschirm unterscheiden, hat der Programmierer von Deluxe-Term alle Register gezogen und ein typisches GEM-Produkt geschaffen. Es wurde aber auch an Benutzer gedacht, die meinen: „Wozu Mausbedienung in einem Terminalprogramm". Bild 3 zeigt ein Help-Menü in dem die Bedeutung von Alternate-Tasten gezeigt wird. Die meisten Funktionen können also auch ohne Maus erreicht werden. Deluxe-Term gibt dem Benutzer noch eine dritte Möglichkeit der Programmsteuerung. In der dritten Bildschirmzeile werden Statusinformationen angezeigt. Durch einfaches Anklicken der Kürzel in dieser Statuszeile können alle angezeigten Einstellungen umgeschaltet werden.

Weitere Extras, die zum Teil absolut ungewöhnlich sind: Xmodem-Protokoll, „Notizblock", Soft-Scrolling, zwei Uhren, Empfangspuffer (editierbar), programmierbare Funktionstasten, zwei verschiedene Zeichensätze umschaltbar – weitere ladbar etc.

Für Besitzer eines Dataphon s21-23d wird Datenübertragung mit Protokoll bei 1200 Baud (halbduplex) möglich. Deluxe-Term schaltet die Koppler automatisch immer im passenden Moment um.

Die Funktionstasten sind in doppeltem Sinn programmierbar. Es lassen sich nicht nur Texte speichern, die dann abgesendet werden, wenn die Taste gedrückt wird. Ein „%"-Zeichen im Funktionstastenstring leitet einen Befehl ein. Befehle sind: Warten bis eine bestimmte Zeichenfolge empfangen wurde, Pause für xxx Sekunden und Auslösen einer anderen Funktionstaste. Insgesamt stehen 20 Funktionstasten zur Verfügung.

Beispiel: Wenn die Tasten „F1" und „F2" folgendermaßen belegt sind

„F1" = %w/login please%f2
„F2" = %p005jackintosh

bewirkt das folgendes: Deluxe-Term wartet bis „login please" empfangen wurde, macht anschließend 0,5 Sekunden Pause und sendet dann das Passwort „jackintosh" aus. Das läßt sich

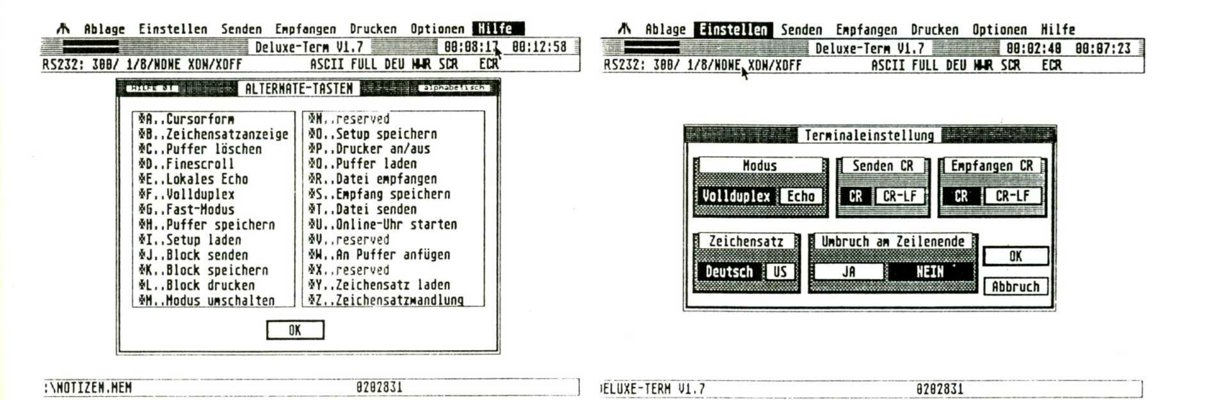


Bild 3: Diese Kommandos sind per Alternate-Taste ausführbar

Bild 4: Terminaleinstellung in Deluxe-Term



Bild 5: Der Notizblock – speicher- und ladbar

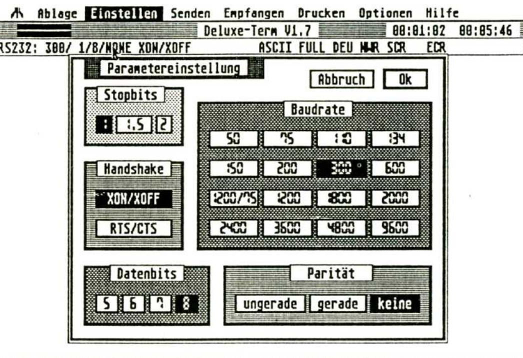


Bild 6: Parametereinstellung in Deluxe-Term

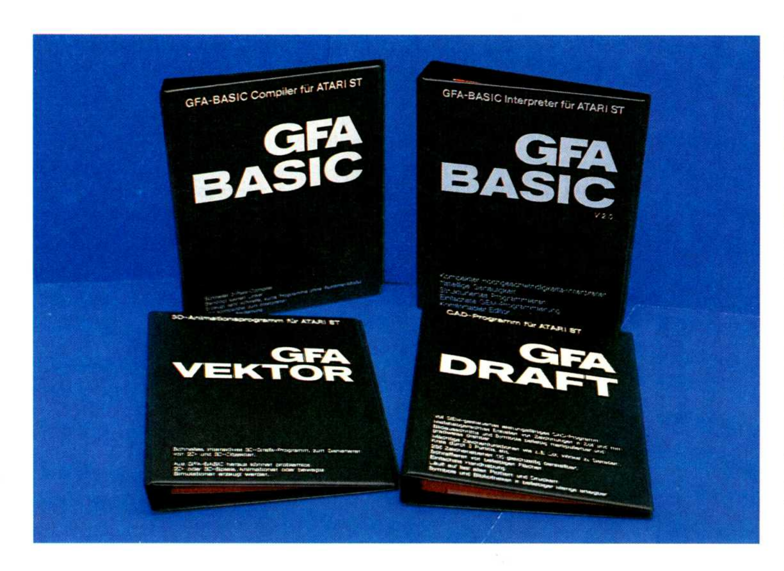
natürlich noch beliebig über insgesamt 20 F-Tasten fortsetzen und damit sollte vollautomatisches Einloggen in jeden Host-Computer möglich sein.

Als einzigen „Nachteil" von Deluxe-Term könnte man den Kopierschutz erwähnen. Hoffentlich ist er gut genug, eine längere Zeit nicht geknackt zu werden. Es wäre ausgesprochen schade, sollten die Entwickler von Deluxe-Term mangels (wirtschaftlichen) Erfolges die Lust verlieren, das Programm weiterzuentwickeln. Geplant sind die Einbindung einiger Terminalemulationen (unter anderem BTX!!!) und das das „Kermit"-Protokoll.

Wenn man bedenkt, daß Deluxe-Term ca. 20 Mark weniger kostet als ST-Term, sollte die Entscheidung keinem potentiellen Käufer schwerfallen.

(Peter Gebhart)

Ein Tag bei der GFA Systemtechnik in Düsseldorf





...Sonntag, 14. Dezember '86, dritter Advent – wir besuchten die GFA Systemtechnik, nicht nur um die neuesten Produkte dieser noch jungen Softwareabteilung zu testen, sondern auch, um ein spannendes Handballspiel live mit zu verfolgen und den Sieger des GFA-Preisausschreibens sowie dessen Programm kennenzulernen.

Die GFA Systemtechnik in Düsseldorf ist eine Zweigniederlassung der GFA in Kiel, welche sich erfolgreich mit Prozeßsteuerung und Automation beschäftigt. Die Düsseldorfer Abteilung befaßt sich vor allem mit der Entwicklung und dem Vertrieb von Software und sollte, nach der rasanten Einführung des GFA BASICs, jedem ATARI ST Anwender ein Begriff sein.

Vor allem zwei Namen prägen das Bild dieser Abteilung, Rolf Hilchner, der sich um einen reibungslosen Geschäftsbetrieb im weitesten Sinne kümmert, und Frank Ostrowski, der Schöpfer des GFA BASIC-Interpreters und des neuen Compilers. Ferner stammen die Programme GFA Vektor (ein Programm zum Erstellen dreidimensionaler Objekte) und GFA Draft (ein CAD Programm mit außergewöhnlichen Eigenschaften) aus dem Hause GFA. Auf den Compiler und das GFA Draft werden wir später ausführlich eingehen.

Computer und Sport

Herr Hilchner, der auch vier Semester Sport studiert hat, bemüht sich nicht nur um gute Softwareprodukte, sondern zeigt auch Interesse am Handballspiel. So sponsert er seit geraumer Zeit den Handball Verbandsligisten **TUS Reuschenberg**. Wir durften beim Spiel um den zweiten Tabellenplatz in der Hammfeldhalle zwischen TUS Reuschenberg und dem SV Neukirchen mit dabei sein. Erfreulicherweise siegte die „Hilcherische Mannschaft" mit 24:21 Toren, trotz der anfänglichen Führung des SV Neukirchen. Die Mannschaft des TUS Reuschenberg ist mir ihren Trikots samt Firmen-Emblem der GFA Systemtechnik und ihrem Sponsor, Herrn Hilchner (ganz rechts) auf Bild 2 zu sehen.

Die besten GFA BASIC Programme

Aber nicht nur Sportliches wurde präsentiert, sondern auch der Sieger des GFA-Preisausschreibens für das beste GFA Basic-Programm. Sieger ist der 23jährige Frank Wolter, der zur Zeit an der Fachhochschule Dortmund Technische Informatik studiert. Für sein



Programm **Movie Star** erhielt er DM 5000,– (Bild 3). Mit seinem 27 KByte langen Programm lassen sich auf einfache Weise Objekte erstellen, die man

sich anschließend in rascher Bildfolge, quasi als Film, auf dem Bildschirm ansehen kann. Dabei ist es nicht erforderlich für jedes Bild des Films ein neues Objekt zu entwerfen, vielmehr stehen dem Anwender viele Effekte zur Verfügung, um ein einmal entworfenes Objekt zu verändern, so daß ein Film, d. h. eine Bewegung entsteht. Hat man z. B. als Objekt einen Würfel entworfen, so kann dieser Würfel nun rotiert, verkleinert oder einfach über den Bildschirm bewegt werden, indem lediglich Start- und Endpunkt bzw. der Drehwinkel festgelegt wird. Im Film werden dann alle Bewegungen (Zwischenbilder) des Würfels zwischen Start- und Endpunkt vom Rechner berechnet, so daß die Objektveränderung als gleitende Bewegung erscheint. Das gesamte Programm läßt sich mit der Maus bedienen.

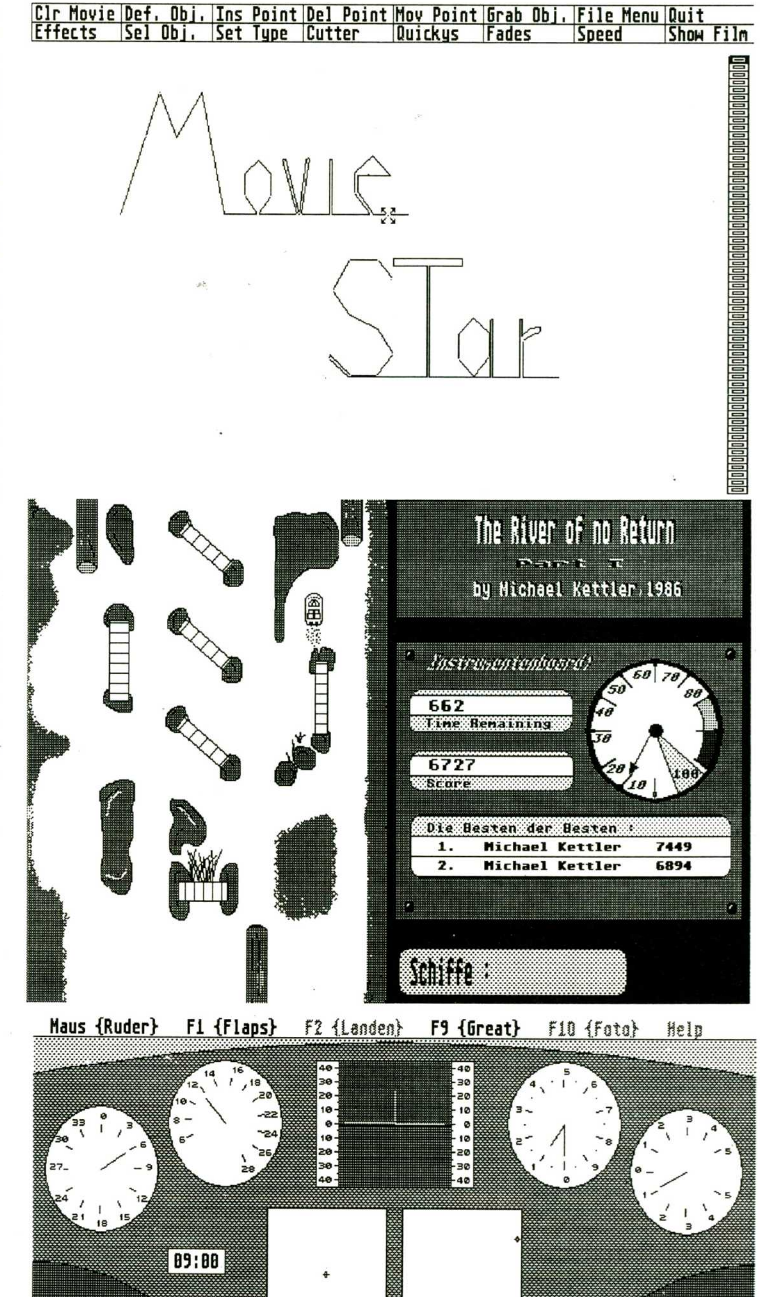
Das zweitbeste Programm ist ein Spiel mit dem Namen **The River of no Return**, das mit DM 1000,– honoriert wurde. Bei diesem Spiel muß ein Motorboot sicher durch einen Flußlauf gesteuert werden, der mit verschiedenen Hindernissen versehen ist. Je länger man sein Boot durch den Fluß steuert, desto schwieriger wird das Spiel und desto mehr Punkte kann man erzielen.

Schließlich ist die Mühe für das drittbeste Programm, einen **Segelflug-Simulator**, mit DM 500,– belohnt worden. Dabei handelt es sich um einen Blindflug-Simulator, bei dem mit diversen Instrumenten und einem künstlichen Horizont geflogen werden muß. Nach einem erfolgreichen Streckenflug mit zuvor gewählten Wendepunkten oder einer Bruchlandung erfolgt eine grafische Auswertung des Fluges.

Alle drei Programme werden von der GFA-Systemtechnik auf einer Diskette für unter DM 80,– verkauft.

Was die Zukunft bringen wird...

Frank Ostrowski wird sich nach einem Jahr intensivster Programmierarbeit (GFA-BASIC und -Compiler) dem Bücherschreiben widmen und dort sein Wissen bekannt geben. Außerdem wird unter seiner Regie ein sehr schneller GFA Assembler für den ATARI ST entstehen, der im März auf der CeBit in Hannover vorgestellt werden soll. Man darf gespannt sein! Aber auch Herr Hilchner schmiedet Pläne, um das GFA BASIC zu einer





Art Weltstandard zu machen. So wird es das GFA BASIC im Juni '87 auch für die IBM AT Rechner geben. An dieser Stelle sei erwähnt, daß das GFA BASIC für den ST schon längst über unsere Grenze hinaus verbreitet und bekannt ist. Mittlerweile gibt's das GFA BASIC auf fast allen Kontinenten der Welt außer dem Ostblock, Asien und Afrika. Allein in Deutschland sind bisher ca. 15 000 Stück verkauft worden und das, obwohl kein Kopierschutz vorhanden ist! Zur Frage nach einem Kopierschutz meint Herr Hilchner, daß ein professionelles Arbeiten (z. B. mit einer Festplatte) generell nur möglich ist, wenn kein Kopierschutz vorhanden ist. Da hilft auch keine einmalige Sicherheitskopie etc. Man könne schließlich nicht erwarten, daß ein Kunde, nach Zerstörung der Original-Diskette, Tage oder Wochen auf Ersatz wartet. Aus diesem Grund wird es auch in Zukunft bei allen Produkten aus dem Hause GFA **kein Kopierschutz** geben, natürlich mit der Hoffnung verbunden, daß der Anwender über Raubkopien und deren negativen Folgen vernünftig zu denken lernt.

GFA BASIC mit hoher Genauigkeit

Daß der GFA BASIC Interpreter für den ATARI ST zu den schnellsten überhaupt gehört, ist wohl unbestritten. Die hohe Geschwindigkeit ist aber unter anderem darauf zurückzuführen, daß der Interpreter keine Variablen mit doppelter Rechengenauigkeit (double-precision) kennt. Die 11-stellige Genauigkeit des GFA BASIC Interpreters kann für manche Anwendungen nicht ausreichend sein. Da nützt auch die hohe Geschwindigkeit nichts mehr. Doch mit der Einführung eines sogenannten Floating-Point Coprozessors, dem MC 68881 für binäre Fließkomma-Arithmetik, werden alle Berechnungen extrem beschleunigt. Einen solchen Hardware-Zusatz wird es z. B. von der Firma Lischka Datentechnik geben. Damit ein solcher Coprozessor auch in BASIC voll ausgenutzt werden kann, wird es ein neues, speziell für diesen Coprozessor überarbeitetes GFA BASIC geben. Natürlich wird dieses schnelle BASIC mit einer hohen Genauigkeit (mind. 19 Stellen) versehen sein. Das neue GFA BASIC mit Floating-Point Unterstützung wird es kurz nach dem Erscheinen mit der erforderlichen Hardware-Erweiterung zum Preis von ca. DM 249,– geben.

Fehler im GFA BASIC V2.0

Obwohl sehr viele Anwender mit dem GFA BASIC Interpreter (V 2.0) arbeiten, sind ein paar kleine Fehler des Interpreters übersehen worden. Dies ist nicht weiter verwunderlich, da Fehler nur bei den folgenden Befehlen unter ganz bestimmten Voraussetzungen auftreten:

– DEFNUM
bei x=0.99999...

– SPRITE X$
ohne Koordinaten und Garbage-Collection

– INPUT
bei mehr als 256 Variablen

– SWAP
bei Bit-Variablen

– LPOS( )
bei PRN:/LPRINT

– VARPTR (A$(X))
bei Berechnungen und als Parameter manchmal falsch

– bei String-Variablen als Parameter bei gleichem Namen und ungerader Länge

Um diese sieben Fehler zu beseitigen, brauchen Sie nur das kleine GFA BASIC-Listing „Patch Version 2.0" abzutippen und zu starten. Anschließend wird Ihre Original GFA-BASIC-Interpreter-(V2.0) Diskette automatisch verbessert, so daß die genannten Fehler nicht mehr auftreten. Bevor Sie das Programm starten sollten Sie eine Sicherheitskopie der Original Diskette anfertigen, damit im Falle eines Tippfehlers die Daten nicht zerstört werden.

Wir begrüßen es, daß die Firma GFA Systemtechnik, Fehler, die wohl in jedem längeren und vor allem so komplexen Programm enthalten sind, bekannt gibt, und gleichzeitig für deren Beseitigung sorgt.

(UB)

```
Rem  Patch Version 2.0
Open "U",#1,"GFABASIC.PRG"
Open "U",#2,"GFABASRO.PRG"
Do
  Read A$
  Exit If A$="END"
  If Len(A$)=2
    Out #1,Val("&H"+A$)
    Out #2,Val("&H"+A$)
  Else
    Seek #1,Val(A$)
    Read A
    Seek #2,A
  Endif
Loop
Seek #1,16903
Out #1,&HBD
Seek #2,16369
Out #2,&H9B

Data  7161, 6661,43
Data  7363, 6863,43
Data  7744, 7244,01,00
Data  7814, 7314,01,00
Data  9770, 9270,E5,41,D6,B0,10,00,61,00,FB,58,4E,AE,FF,40
Data 10630,10130,D6,FC,00,0C,52,8D,4E,75,52,40,E2,48,54,97,4E,75
Data 27866,26084,3C,00,52,88,22,5F,60,00,FF,5A
Data 35196,33414,4E,AE,FF,48
Data 36006,34224,BC,40,64,AE,BE
Data 36220,34438
Data 60,04,00,00,00,00,24,5F,34,12,34,80,26,62,27,4A,00,4A,04,42
Data 00,4E,D6,C2,48,C2,44,82,27,42,00,4E
Data 36241,34459,4E,D6,C2,48,C2,44,82,27,42,00,4E
Data END
Rem
```

Klick mich

Utilities aus der Schublade



Eine äußerst wichtige Eigenschaft von GEM, ist die Verwaltung von Deskaccessoirs. Sie sind jederzeit bereit, um mitten in einem Programm aufgerufen zu werden (im Volksmund auch 'Anklicken' genannt) und kehren nach verrichteter Arbeit wieder zum eigentlichen Programm zurück. Drei Programme, die auf dieser Basis arbeiten, wollen wir uns hier zu Gemüte ziehen.

Die Ideen, die in solchen Programmen stecken, sind sehr vielseitig. Der Grundgedanke ist es, oft benötigte Hilfsprogramme aufrufen zu können, ohne das laufende Programm zu verlassen. Dazu gehören beispielsweise Terminkalender, Uhr, Wecker oder etwa ein Notizblock. Wäre es nicht praktisch, wenn man automatisch an einen bestimmten Termin erinnert würde oder zum schnellen Errechnen des Lottogewinns mal eben einen Taschenrechner aus dem Rolladen ziehen könnte. Unsere drei Testprogramme beinhalten solche Hilfsprogramme und nehmen dem Benutzer einige Arbeit ab.

Cornerman

Das erste, was an Cornerman auffällt, ist die rechts oben auf dem Bildschirm erscheinende Uhrzeit. Sie tickt nicht nur in GEM-Programmen, sondern immer und überall. Wem die digitale Anzeige nicht gefällt, kann zusätzlich eine Zeigeruhr aufrufen, die die Zeit in altüblicher Art und Weise darstellt. So harmlos diese Uhr auch ausschaut, sie hat einiges zu bieten. Vergrößert man sie über den gesamten Bildschirm ersetzt sie voll und ganz eine herkömmliche Küchenuhr und auch aus der Ferne ist sie lesbar. Doch dem nicht genug, versucht man sie wieder zu verkleinern wird man feststellen, daß dies nicht ohne weiteres geht. Hinter der Uhr steckt nämlich noch ein schöner Schutz, da die Uhr erst dann verschwindet und damit das normale Programm freigibt, wenn man ein Passwort eintippt. Zwar wüßte ich momentan nicht, vor wem ich etwas schützen sollte, aber wer weiß?

Neben einem Notizblock, der sich, wie sollte es auch anders sein, zur Aufnahme von Notizen eignet, wurde eine ASCII-Tabelle eingebaut, die jederzeit die ASCII-Codes sichtbar macht. Besonderen Wert legte der Autor auf eine automatische Telefonwähleinrichtung, die auf Befehl die zu einer Adresse gehörende Telefonnummer wählt und somit das lästige Nummerntippen erübrigt. Schade ist natürlich, daß dies nicht ohne zusätzliche Hardware funktioniert, auch die Nutzung als Adressdatei ist aufgrund der vorgegebenen Nummernmaske, die hierzulande unpassend ist, fraglich.

Natürlich lassen sich die verschiedenen Daten auch ausdrucken und zwar

auch auszugsweise. Leider wurde auf eine Druckeranpassung für eventuelle Sonderzeichen verzichtet.

Desk-Assist II +

Wie alle Programme dieser Klasse, bietet auch dieses einige Besonderheiten. Viel Wert legte der Programmierer auf die Druckerfeatures. Als erstes wäre da der Druckerspooler zu erwähnen. Er druckt bis zu fünf Dateien von Diskette im Hintergrund, ohne das Hauptprogramm groß zu behindern. Die Tatsache, daß die meisten Spooler nur reine ASCII-Dateien drucken können, trifft hier nicht zu. Durch eine eigene Druckeranpassung, die gewisse Ähnlichkeiten mit der 1st__Word-Anpassung aufweist, werden auch verschiedene Schriftarten und Formatanweisungen akzeptiert, sodaß dieser Spooler auch 1st__Word Texte zu Papier bringt.

Hardcopy gespoolt????

Eine weitere Fähigkeit dieses Programmes ist eine eigene Hardcopyroutine, die entweder den gesamten Bildschirm zu Papier bringt oder nur (oder sogar) einen definierten Bildschirmbereich. Hierzu wird der zu druckende Bereich per Maus definiert und auf die Drucknadeln übertragen. Das Besondere daran ist, daß sich die zur Hardcopy nötigen Steuercodes, also der Grafik-Modus des Druckers und der Zeilenabstand frei definieren lassen, was Besitzer bestimmter Drucker sicher freut. Die Hardcopy läßt sich aber auch auf Diskette umlenken und kann zu einem anderen Zeitpunkt direkt von Diskette auf den Drucker gebracht werden – gespoolt versteht sich.

Wer aus irgendwelchen Gründen seine Daten, Programme oder sonstigen Files vor unbefugtem Gebrauch schützen will, der findet in Desk-Artist eine wirkliche Hilfe. Jedes beliebige File, sei es ein C-Listing, ein Compiler oder 1st__Word, wird mit einem Schlüsselwort verknüpft und somit in Byte-Salat verwandelt. Durch das gleiche Schlüsselwort entsteht wieder Klartext.

Natürlich sind die Druckerbefehle nur ein Teil dessen, was ein Desk-Utility beherrschen sollte. Wie auch bei den anderen Programmen wird auch hier Wert auf Notizblock und Terminkalender gelegt, wobei gerade der Terminkalender recht schön gelungen ist. Neben den Wochentagen werden auch Feiertage, da es sich dabei um besonders wichtige Tage im Jahr handelt, angezeigt und zwar mit dem entsprechenden Namen. Zudem kann jeder Termin mit einem akustischen Signal hervorgehoben werden.

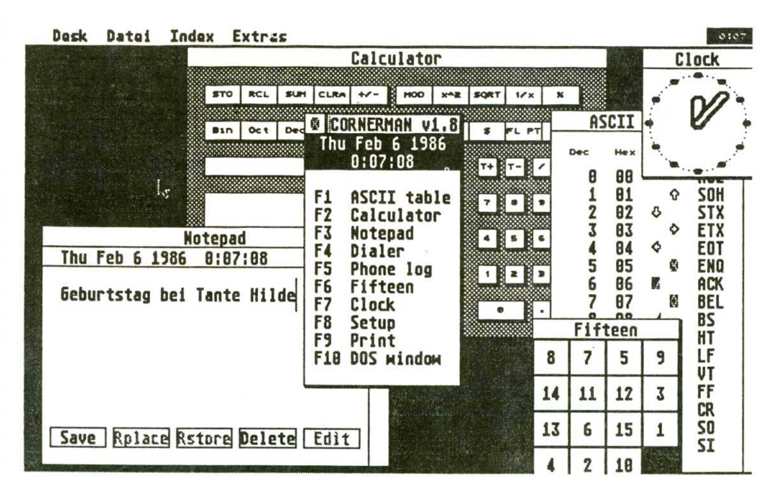


Bild 2: Cornerman: Eine Übersicht

Als einzigstes der drei Programme, bietet Desk-Artist eine Adressdatei, welche zwar nicht mit einem reinen Adressprogramm mithalten kann, aber die wichtigsten Fähigkeiten (Suchen, Drucken) beherrscht.

Als letztes sei der implementierte Taschenrechner erwähnt. Er bietet neben den Grundrechenarten eine sehr umfangreiche Umrechnung von Maßeinheiten, hat aber ansonsten nicht viel zu bieten.

Alles in allem bietet DESK-Assist viele wirklich nützliche Fähigkeiten, die dieses Programm interessant machen.

Sideklick

Das letzte Programm im Bunde ist sicherlich der bekannteste Vertreter dieses Trios. Dies liegt sicherlich auch daran, daß Sideklick das älteste Werk,

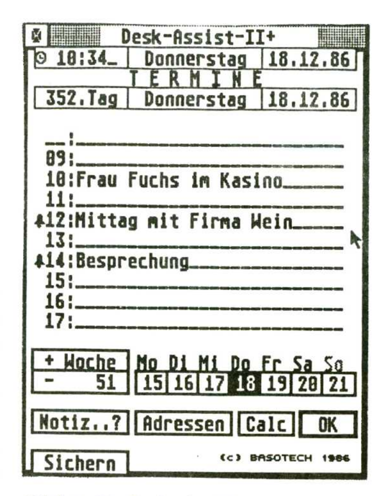


Bild 3: Desk Assist II:

Der Terminkalender

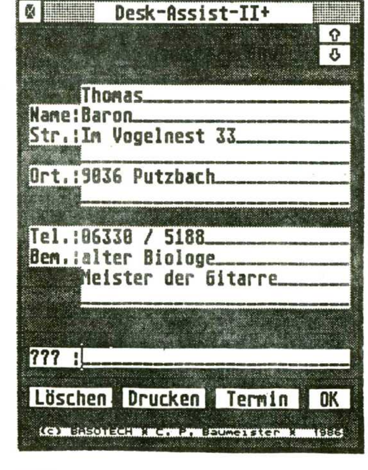


Bild 4: Desk Assist II:

Die Adressdatei

Willkommen zu Side-Click V2.2!
Taschenrechner
Terminkalender
Kalender
Uhr
Wecker
Notizblock
Zeichensatz
Drucke Disk-Index
Drucker-Anpassung
31.12.86
23.59 Uhr
Stellen
Abbruch

Bild 5: Sidekick: Das Menü

mit noch dazu dem originellsten Namen ist. Sideklicks Repertoire beinhaltet Notizblock, Terminkalender, Uhr, Wecker, ASCII-Tabelle, Taschenrechner, einen normalen Kalender und ein Directory-Druckprogramm.

Die Uhrzeit erscheint nach Wunsch in einem winzigen Fenster, welches nach Belieben auf dem Bildschirm plaziert werden kann. Der zugehörige Wecker erinnert an wichtige Termine außerhalb der Rechnerwelt, die ansonsten nur zu leicht vergessen werden. Man denke nur an nächtliche Tennis-Live-Übertragungen.

Der Terminkalender ist ein nützliches Hilfsmittel, das dafür sorgt, daß langfristige Termine nicht vergessen werden. Die 'Suchfunktion' übernimmt das lästige Blättern und hilft bei der Auffindung bestimmter Termine.

Der Ausdruck dieser Termine oder auch anderer Daten wird durch eine leicht bedienbare Druckeranpassung unterstützt Probleme bei Sonderzeichen und Schriftarten treten somit nicht auf.

Taschenrechner inclusive

Ein Bestandteil, der in jedem Paket enthalten ist, ist der Taschenrechner. Er scheint so wichtig, daß keiner der Programmautoren darauf verzichtete. Die Qualitäten der Rechner weisen aber gewaltige Unterschiede auf. Während die Rechner von Cornerman und Desk-Assist nur die nötigsten Operationen beherrschen, bietet Sideklicks Rechner einen wirklich leistungsfähigen Taschenrechner. Er arbeitet mit UPN-Notation, also mit der Rechenweise, mit der die HP-Taschenrechner (Hewlett Packard) ausgestattet sind. Hat man noch nie mit einem solchen Rechner gearbeitet, ist es etwas ungewöhnlich, nach ein wenig Übung merkt man aber schnell, daß diese Methode sehr effektiv ist und gegenüber der 'normalen' AOS-Methode viele Vorteile bietet. Um es kurz zu machen: Sideklicks-Rechner ersetzt ohne Probleme einen guten Taschenrechner, dessen Preis den von Sideklick fast überbieten dürfte; nur eben doch etwas handlicher ist.

Fazit

Das Prinzip aller drei Programme ist gleich. Der Unterschied liegt in den eingebauten Fähigkeiten, die je nach Geschmack das eine oder andere Programm favorisieren. Desk-Assist sticht durch die Adressdatei, den Terminkalender und den (Hardcopy)-Spooler hervor. Sideklick glänzt durch den wirklich professionellen Taschenrechner, die Terminverwaltung und den Ausdruck des Disketteninhaltes. Cornermann ist etwas im Nachteil, zumindest für den deutschen Markt. Eine angepaßte Version würde, vor allem für den DFÜ-interessierten Benutzerkreis, ein äußerst interessantes Programm darstellen.

Die Zuhilfenahme solcher Utilities ist meist recht praktisch, bei manchen Anwendungen erfüllen aber auch herkömmliche Hilfsmittel (z. B. Notizblock, Adressbuch) ihren Zweck.

(HS)

	Cornerman	Sideklick	Desk-Assist
Taschenrechner	+	+	+
Notizblock	+	+	+
Adressbuch	–	–	+
Terminkalender	–	+	+
Kalender	–	+	+
Uhr	+	+	+
Wecker	–	+	+
Directorydruck	–	+	–
Druckeranpassung	+	+	–
ASCII-Tabelle	+	+	–
Autowähler	(+)	–	–
Verschlüsselung	–	–	+
Druckerspooler	–	–	+

Bezugsadressen:

Cornerman: div. Distributoren

Desk Assist 2+: Basotech
Johann Henk Str. 18
5307 Pech bei Bonn

Sideklick : RDS Software
Jakobstr. 8a
6096 Raunheim

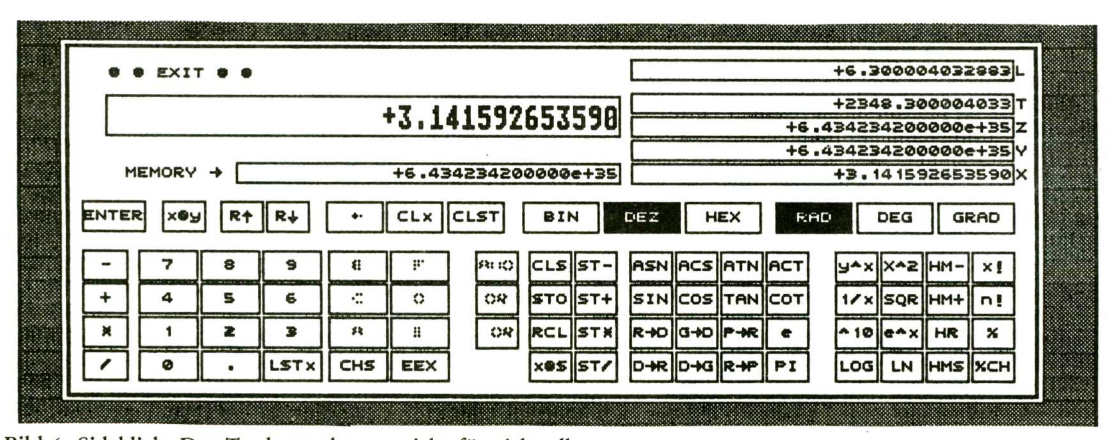


Bild 6: Sideklick: Der Taschenrechner spricht für sich selbst

Es muß nicht immer ein Interpreter sein
Der Compiler zum GFA-BASIC

Man hat einen sehr schnellen BASIC-Interpreter, der bequem zu bedienen ist, einen mächtigen Befehlssatz hat und sehr kompakt ist. Läßt sich eine solche Sprache noch erweitern? Dafür gibt es jetzt einen Compiler: Wir testen für Sie den brandneuen Compiler für den GFA-BASIC-Interpreter.

Der GFA-BASIC-Interpreter ist nicht nur wegen seiner Struktur eine BASIC-Revolution (keine Zeilennummern, Programmstruktur, etc.), sondern auch wegen der für einen Interpreter ungewöhnlich schnellen Ausführung. Von einem Compiler verspricht man sich in der Regel eine Geschwindigkeitssteigerung beim Ablauf eines Programms. Das ist jedoch nicht der einzige wichtige Faktor: Ein BASIC-Programm wird mit dem Interpreter geschrieben und getestet, dann wird es compiliert – und schon braucht man den Interpreter nicht mehr. Zu dem Geschwindigkeitsgewinn kommt hinzu, daß man ein BASIC-compiliertes Programm einfach von DESKTOP aus starten kann, ohne den Interpreter vorzuladen.

GFA BASIC Compiler 1.0

Stoppen | Immer | Schleife | Nie

Trapv | + | -

Errors | Text | Zahl

Bomben | + | -

Compiliere | Abbruch

Compilierungszeit eines 56 KByte großen Files		
Floppy	Ram-Disk	Hard-Disk
95 Sek.	40 Sek.	42 Sek.

Tabelle 1

Der freundliche Compiler

Der GFA-Compiler ist ein 2-Pass-Compiler, der BASIC-Programme, die mit dem GFA-BASIC-Interpreter geschrieben sind und die mit SAVE abgespeichert wurden, in Maschinenprogramme umwandelt. Ein integrierter Linker sorgt für die Einbindung der nötigen Routinen. Tatsächlich werden nur die benötigten Routinen eingebunden, so daß die Programme nicht unnötig lang werden. Die Bedienung des Compilers ist ebenso einfach wie die des Interpreters. Nach dem Start meldet sich ein Fenster mit den Einstellparametern. Man kann verschiedene Parameter anklicken, zum Beispiel, ob der Abbruch mit Break erlaubt ist oder nicht, das Aussehen der Fehlermeldung oder andere. Nach dem Anklicken des gewünschten Files wird das Programm compiliert. Der Compilier-Vorgang läuft recht flott; wir empfehlen trotzdem die Anwendung einer Harddisk oder einer Ramdisk. Die Handhabung des Compilers ist so einfach gehalten, daß man bereits nach kurzem Lesen des mitgelieferten Handbuchs fertige Programme compilieren kann.

In Sachen Geschwindigkeit

Wie erwähnt, ist der BASIC-Interpreter sehr schnell. In vielen Fällen, vor allem bei den Graphik-Funktionen, ist dieser Interpreter schneller als mancher Compiler. Deshalb ist nicht in allen Fällen eine bedeutende Geschwindigkeitssteigerung zu erzielen.

Viele Graphik-Funktionen des Interpreters (etwa RBOX, POLYLINE oder PBOX) rufen fertige Routinen des VDI-Systems auf, so daß sie außerordentlich schnell laufen. Das Füllen des gesamten Bildschirms durch den Befehl FILL ist bei dem Interpreter ebensowenig meßbar wie beim Compiler. Versucht man jedoch, den Bildschirm durch POKE zu füllen (was allerdings nicht sehr sinnvoll ist), dann ist der Interpreter sehr langsam. Der gleiche Versuch mit einem compilierten Programm ergibt einen Geschwindigkeitsgewinn um den Faktor 3 bis 4, wie aus der Tabelle 2 zu ersehen ist.

Bei mathematischen Funktionen ist die Steigerung deutlicher. Wir haben die Benchmark-Tests, die auf der Interpreter-Diskette abgespeichert sind, compiliert und die Ergebnisse mit dem interpretierten Ablauf verglichen. Das Resultat entnehmen Sie bitte der vorliegenden Tabelle.

Die **Neuen** und die **Erfolgreichen aus dem Heim-Verlag**



B-407 Buch 49,– DM
D-427 Diskette 39,– DM

Wenn Sie das Software-Paket **VIP-Professional** kaufen wollen oder schon besitzen, dann weiht Sie dieses Buch schnell und umfassend in die Geheimnisse dieses Profiprogrammes ein.

VIP-Professional besteht aus den drei Funktionsbereichen

- DATENBANK
- KALKULATION
- GRAFIK

mit denen wichtige und vielfältige Aufgaben hervorragend gelöst werden.

Das Buch enthält komplette Musterlösungen für die Gewinn- u. Verlustrechnung und Fakturierung. Anhand dieser Beispiele wird gezeigt, wie Sie alles herausholen, was in VIP-Professional steckt.

Mit diesem Buch können Sie **VIP-Professional** richtig einsetzen und seine Möglichkeiten voll ausschöpfen.

Anhand von über 130 Programmbeispielen und Erläuterungen lernen und trainieren Sie das Programmieren in BASIC. Von einfachen, aber grundlegenden Beisp. bis zur ausgereiften Anwendung findet der ATARI ST-Besitzer Beispiel-Programme, die den perf. Einstieg in die Programmiersprache des GfA-BASIC's leicht machen. Alle Beisp. werden so dargestellt, daß das Verständnis für die Programmstruktur gefördert wird, aber auch die Details der GfA-BASIC-Version deutlich werden. Die Progr. sind gut dokumentiert, wobei die hervorr. Möglichkeiten des GfA-Basic zur strukturierten Programmierung genutzt werden. **Aus dem Inhalt:** ● Eigenschaften des GfA-BASIC ● Dateiverwalt. u. Ordner in GfA-BASIC ● Schleifentechnik ● Felder ● Unterprogramme ● Menü-Steuerung ● Window Technik ● Zufallszahlen ● Seq.- u. RANDOM-Dateien ● Textverarbeitung ● Sortierprogramme ● Fakturiersysteme ● Grafik ● Spiel uvm.

Zum Buch: Programmdiskette mit sämtl. Beispielprogr. 320 Seiten mit 131 Programmen in GfA-BASIC

ATARI ST
Prof. Dr. B. Bollow / K. Reimann
GfA-BASIC
Programm-Sammlung
131 Programme in GfA-BASIC

B-408 Buch 49,– DM
D-428 Diskette 49,– DM

Das Buch für den richtigen Einstieg mit dem ATARI ST. Leicht verständlich wird der Lernende in den Lernstoff eingeführt.

Einige der Themen:

Die Hardware des ATARI ST

- Überblick über die Systemkomponenten und ihre Funktionsweise
- Aufstellung und Wartung des Computers

Die Software des ATARI ST

- Arbeiten mit dem GEM (das Desktop/Maus/Icons etc.)
- Kopieren von Files und Disketten, Löschen und Formatieren u.v.a.m.
- Die Programmiersprachen BASIC und LOGO
- Einführung in die Bedienung und Programmierung
- Programmsammlung mit vielen interessanten Beispielen

350 Seiten mit 55 Abbildungen.



B-400 Buch 49,– DM
D-420 Diskette 39,– DM



B-402 Buch 49,– DM
D-422 Diskette 39,– DM

Der ideale Einstieg in die Programmierung mit LOGO. Ein Buch für alle, die LOGO schnell verstehen und perfekt erlernen möchten. Ein Buch wie es sein muß, leicht zu verstehen und interessant geschrieben.

Einige der Themen:

- **Variablen- und Listenverarbeitung**
- **Packages und Properties**
- **Die Turtlegrafik**
- **Viele ausführlich erklärte Beispielprogramme zu den Themen**
  - Arbeiten mit relativen und sequentiellen Dateien
  - Anwendungen aus Mathematik
  - Hobby- und Spielprogramme u.v.a.m.

Zum Buch gibt es die Programmdiskette mit sämtlichen Beispielprogrammen.

ca. 350 Seiten mit vielen Abbildungen.

Ein Grundwerk für alle ATARI ST-Besitzer. Auf über 300 Seiten eine klare und verständliche Einführung in die Programmiersprache BASIC, elementare BASIC-Kommandos, Diskettenhandhabung und vieles, was zur perfekten Beherrschung des ATARI ST gehört.

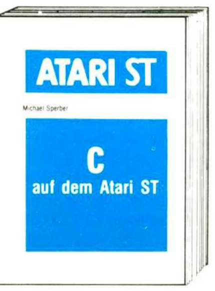
Ein Spitzenbuch mit über 80 Übungs- und Anwenderprogrammen wie z. B.:

Sortierprogramme / Textverarbeitung / Umgang mit sequentiellen- u. Random-Dateien / Fakturierprogramm / Programmiertechniken an ausgewählten Beispielen u. v. a. m.

Zum Buch gibt es die Programmdiskette mit sämtlichen Beispiel-Programmen.



B-401 Buch 49,– DM
D-421 Diskette 39,– DM



B-406 Buch 49,– DM
D-426 Diskette 39,– DM

C ist die zweite „Muttersprache" des Atari ST: schnell, komfortabel, kompakt im Code.

„C auf dem Atari ST" ist für Anwender geeignet, die Erfahrungen mit anderen Programmiersprachen gemacht haben. Das Buch behandelt den vollen Sprachumfang von Standard-C und verweist auf BASIC und Pascal.

Besonderer Wert wird auf die Anschaulichkeit und Genauigkeit der Darstellung gelegt. Alle Programmbeispiele sind getestet und direkt in den Text übernommen.

Aus dem Inhalt: C-Compiler für den Atari: Digital Research, Lattice, Megamax ● der Editor ● Bedienung des Compilers ● Grundlegende Elemente eines C-Programms ● Variablentypen ● Felder und Vektoren ● Ausdrücke ● Zeiger ● Speicherklassen ● Bitfelder ● Varianten ● Aufzählungen ● Dateien ● Diskettenhandling ● Einbindung von Assemblerprogrammen ● Bildschirmgrafik in C ● Fehler in den C-Compilern ● Tools u. a.

Ein Grundwerk.

Wer sich mit der Programmierung der GEM-Funktionen vertraut machen will braucht dieses Buch!

Es beginnt mit einer Erläuterung des GEM-Aufbaus und führt anhand von Programmbeispielen zum **leichten Verständnis aller auf dem Atari verfügbaren GEM-Funktionen.** Die übersichtliche Gliederung ermöglicht auch ein schnelles Nachschlagen der Funktionen.

Einige der Themen:

● Was ist **GEM** ● Die GEM-Bestandteile **VDI** und **AES** ● Die **GEM-Implementation** auf dem Atari ST ● Aufruf der GEM-Funktionen aus **BASIC, C,** und **ASSEMBLER** ● Sonstige **Programmiersprachen** und GEM ● Die Programmierung der **VDI**-Funktionen ● Die **AES**-Bibliothek und die Programmierung ihrer Funktionen ● Aufbau eines **Objektbaumes** ● Was ist eine **Resource**-Datei? ● Viele erläuterte **Beispielprogramme** in BASIC, C und ASSEMBLER.



B-404 Buch 49,– DM
D-424 Diskette 39,– DM



B-405 Buch 49,– DM
D-425 Diskette 39,– DM

Zum neuen Basic Interpreter, ein Buch, das mit gezielten Beispielen verständlich den Einstieg in das Basic der Superlative ermöglicht. Ein Muß für jeden Besitzer dieses Interpreters. Einige der Themen:

- Der Umgang mit dem Editor
- Ausführliche, mit Beisp. versehene Befehlsübersicht
- Die fantastischen Grafikmöglichkeiten (Windows, Sprites, Alertbox, Pull-down Menüs
- Strukturierte Programmierung – auch in Basic möglich!
- Dateiverwaltung unter GfA-Basic
- GEM Handhabung in Basic
- Vergleich mt anderen Basic-Interpretern – Hilft bei der Kaufentscheidung
- Zahlreiche Übungs- und Anwenderbeispiele
- Mathematik und hohe Genauigkeit

Zum Buch gibt es die Programmdiskette mit sämtlichen Beispielprogrammen.

ACHTUNG: Zu den Büchern gibt es die Programmdiskette mit allen Programmbeispielen – Diskette laden und los geht's...



6100 Darmstadt-Eberstadt
Heidelberger Landstr. 194
Telefon 0 61 51 / 5 60 57

▶ **Für Bestellungen verwenden Sie am besten die in der ST-Zeitschrift vorhandene BUCH- UND SOFTWARE-BESTELLKARTE**

▶ **Bücher und Programm-Disketten aus dem Heim-Verlag erhalten Sie auch bei Ihrem ATARI-Fachhändler oder im Buchhandel**

*alle Preise sind unverbindliche Preisempfehlungen

Auch eine Sortier-Routine, die auf dem Interpreter sehr langsam läuft, wird in der compilierten Version recht schnell. Freilich ist der Geschwindigkeitsfaktor nicht allein von dem Compiler abhängig. Wenn schon im Interpreter umständlich programmiert wird, kann auch der beste Compiler den Code nicht optimieren, und das fertige Programm wird nicht wesentlich schneller. Da ist zum Beispiel der Fall der Nutzung von Integerarithmetik. Da der Compiler die Integerarithmetik effizient benutzt, ist zu empfehlen, immer dort, wo ganzzahlige Variablen benötigt werden, nur Zahlen des Typs Integer zu verwenden.

Bildschirm füllen durch direktes Poken	
Mit For...Next Schleife	
Interpreter	Compiler
Mit Integer : 10,96 Sek. Mit Fließk. : 13,27 Sek.	Mit Integer : 3,55 Sek. Mit Fließk. : 4,33 Sek.
Mit Repeat...Until Schleife	
Interpreter	Compiler
Mit Integer : 21,51 Sek. Mit Fließk. : 23,02 Sek.	Mit Integer : 3,38 Sek. Mit Fließk. : 6,03 Sek.

Tabelle 2

Wie groß wird ein compiliertes Programm?

Die Programme in GFA-BASIC sind in der Regel nicht gerade klein. Ein compiliertes Programm wird noch länger. Man darf nicht vergessen, daß der Compiler viele Routinen einfügen muß. Auch hier spielt die Art und Weise, wie man in BASIC programmiert, eine große Rolle. Bei kürzeren Programmen ist, wie bei allen Compilern, das Resultat etwas schockierend. Etwas umfangreichere Programme dagegen werden nicht übermäßig groß.

BENCHMARKS		
Benchmark	Interpreter	Compiler
1	0,04 Sek.	0,02 Sek.
2	0,042 Sek.	0,015 Sek.
3	1,17 Sek.	0,215 Sek.
4	1,08 Sek.	0,215 Sek.
5	1,16 Sek.	0,24 Sek.
6	1,585 Sek.	0,35 Sek.
7	2,16 Sek.	0,475 Sek.
8	3,08 Sek.	2,17 Sek.

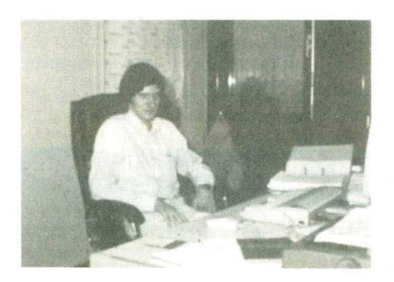
Tabelle 3

Über den Autor

Entwickler des GFA-BASIC-Interpreters und des Compilers ist Frank Ostrowski. Mit 26 Jahren ist er schon eine Legende, wenn es um Software-Entwicklung im deutschen Raum geht. In rund einem Jahr schrieb er Interpreter und Compiler. Auf seinem Schreibtisch bei der GFA-Systemtechnik wimmelt es von abgedruckten Source-Code-Listen seines GFA-Compilers. Trotz seiner unbestreibaren Fähigkeiten und dem verdienten Ruhm bleibt er gelassen bei seiner Arbeit. Mit Sicherheit werden ATARI-Besitzer noch mehr von diesem jungen Programmierer hören.

Länge eines Files		
File	Interpreter	Compiler
Beispiel 1	264 Byte	4339 Byte
Beispiel 2	1370 Byte	8747 Byte
Beispiel 3	56409 Byte	86740 Byte

Tabelle 4



Fazit

Der GFA-BASIC Compiler ist sicher eines der interessantesten Produkte, die es für den ATARI ST gibt. Für alle BASIC-Programmierer, die ihre Programme vermarkten möchten, ist dieser Compiler ein Muß. Für diejenigen, die nur für sich selbst programmieren, bietet er den Vorteil, daß nach Erstellen und Testen eines Programms der Interpreter nicht immer geladen werden muß. Wir können allen BASIC-Programmierern dieses Programm mit gutem Gewissen empfehlen. Außerdem sind Interpreter und Compiler ohne weiteres als BASIC-Entwicklungspaket geeignet.

(MM)

```
Alert 1,"Kilobaud Benchmarks",1,"Return",A
Lprint "Benchmarks GFA-BASIC Compiler  "
Lprint
Dim M%(5)
Lprint "Benchmark 1 :"'
T%=Timer
For I%=1 To 1000
Next I%
Lprint (Timer-T%)/200
Lprint "Benchmark 2 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
Until K%>999
Lprint (Timer-T%)/200
Lprint "Benchmark 3 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%/K%*K%+K%-K%
Until K%>=1000
Lprint (Timer-T%)/200
Lprint "Benchmark 4 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%/2*3+4-5
Until K%>=1000
Lprint (Timer-T%)/200
Lprint "Benchmark% 5 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%/2*3+4-5
  Gosub Dummy
Until K%>=1000
Lprint (Timer-T%)/200
Lprint "Benchmark 6 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%/2*3+4-5
  Gosub Dummy
  For L%=1 To 5
  Next L%
Until K%>=1000
Lprint (Timer-T%)/200
Lprint "Benchmark 7 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%/2*3+4-5
  Gosub Dummy
  For L%=1 To 5
    M%(L%)=A%
  Next L%
Until K%>=1000
Lprint (Timer-T%)/200
Lprint "Benchmark 8 :"'
T%=Timer
K%=0
Repeat
  K%=K%+1
  A%=K%^2
  B%=Log(K%)
  C%=Sin(K%)
Until K%>=1000
Lprint (Timer-T%)/200
Procedure Dummy
Return
```

BIETE HARDWARE

Kleinanzeigen

■ DISKETTEN ■
■ 5 1/4", 48 tpi, **DM 0,99**, 2D ■
■ 3 1/2", 135 tpi, **DM 3,19**, 1DD ■
■ 3" Schneider **DM 5,85** ■
■ auch andere, bes. Garantie ■
■ Allg. Austro-AG, Ringstr. 10 ■
■ D-8057 Eching, Tel: 08133/6116 ■

SF354 DM 150,– Tel. 09771/5434

SF 354 VB 220,– 02736/6909

Verk. LWST314 DM 600,– VHB
und LWST354 DM 200,– VHB
Wichmann, Tel. 0621/552865

SF354 150,– T. 08092/5565 ab 18⁰⁰

---

260 ST, 1 MB, Roms, Ct-Uhr, Maus
SM 124. CUMANA Doppelfloppy
Erst 10 Mon. alt. 40 % unter NP
Kurt Stückle Tel.: 07127/31326

Thomson RGB Colormonitor
CM 36512 AR für ST: 500,– DM
R. Bubser, Großer Kamp 63
3170 Gifhorn, Tel: 05371-50936

260ST, 1Mb, im IBM Geh. + 130W
Netzt. mit Ven. + getr. Tastat. +
S/Wmon. + Farbm. + 3 1/2" Doppellaufw. je 720K + 5 1/4" Laufw.
(umschaltbar) + Handbücher +
Software zu verk. Preis VS.
Tel. 07402/370 n. 19 Uhr

SF 314 für nur 500 DM zu verk.
(wenig gebraucht) 02689/1478

Verkaufe: GX 80 Epson Drucker
neu mit Atari oder IBM
Interface + Traktor + Farbband VB
800,– DM Tel. 0234/512505

Thomson Farbmonitor, neuwertig
FP 900,– **DM** Tel.: 06401/7453

---

520 ST+, incl. Maus, Monitor
SM124, Floppy SF314, 2100 DM
Köln 80, Tel. (0221) 668413

SF354, DM 230, Tel. 02175/4105

SF354 mit Garantie VB 190 DM!!!
Dirk Stamm, Tel. 0202/471305!!!
★Vip Buch★ 30 DM★(Spitze)

Floppy SF 314 449 DM, Original-Prgs: Textomat, Profimat, Deep
Space je 45 DM. Degas 69 DM.
dBMAN dt. 249 DM. VIP-GEM
449 DM, Maschke, 8601 Reckendorf, Friedenstr. 14, 09544/5272

---

520 ST+ 1MB, Monitor (SM124),
Floppy (SF354), Maus und ROM's
gegen bar (DM 1700,–) sofort abzugeben. Tel. 0711-573864

Eprommer mit Textool nur 145,–
Accu-Uhr wie St-Uhr nur 45 DM
Info 1 DM Tel. 040-862771

NLQ-Drucker, f. AT-ST, Eps.-komp., DM 400, 069/880257

Verkaufe Floppy (neu)
3 1/2" 720 KB komplett mit Gehäuse und eingebautem Netzteil für
DM 450 Tel. 0911/717984-718631

Verkaufe: Atari 260ST + Maus +
Floppy SF354 + Thomson Farbmonitor + MCC Lattice C-Compiler DM 2000,– Tel. 0711/703733

SF354 neuw. 170,– Tel. 07732/6509

Atari 520ST+ SM124, SF314 und
Zubehör VB 2200 Tel. 02522/4924

SF354 8/86 VB 320,– 02251-61134

SF 354 – 2 Monate alt
VB 200,– Tel.: 08191/50462

SF354 neuw. VB 250,– 02833/7374

SF-314 VB: 500 DM
Tel. ab 17–21 Uhr: 07131/72639

SUCHE SOFTWARE

Cobol-Compiler ges. 02233/32740

Suche 24 Track V1.1 0571/77898

Suche für ST, gute + leistungsfäh.
Softw., Textverarb., Karteiverwalt.
u. ä. geeignet für Handelsvertretung
Tel. 07476/472

Treiberprogramme für Gabi 9009
1st-Word für versch. Zeilenabstände, Schreibschritte 10, 12, 15, PS etc.
gesucht. Tel: 0228/216127

Suche Programme zur Finite-Element-Methode. F. Kaufmann, Bismarckstr. 91, 51 Aachen 0241/541777

Gesucht: f. CP/M! Aber auch f.
GEM. Wagner, Zwinglipiatz 3
CH-8001 Zürich

Suche **GOAE-82-Privatliquidation**
mögl. in GfA-Basic für Atari 520+.
Tel. 0931/409800

Suche gewerbl. nutzbare Software
wie Textv., Kalkulation, Datenbank
ect. Fibu mit Anleitung
H.-J. Metten, Dorfstr. 2
5952 Attendorn-Helden
02722/8124

SUCHE leistungsfähiges, KBV-zugelassenes **Arztprogramm** für ST
(mehrplatzfähig, netzwerkverb.)
Dr. W. Allmeling, Internist, 2057
Wentorf, Hauptstr. 3, 040/7205151

Textsanner (Programmierer)
gesucht für ST 520+
Tel: 04103-2248

KONTAKTE

ATARI ST Kontakte gesucht
Tel. 05171/3723

BIETE SOFTWARE

★ Drucker_NL-10_Treiber ★
★ für 1st_Word (+) ca. 100 ★
Schrift. m. ausf. Anl. 20, – auch
★ für Eps.u.kompat. K. Plüher★
★ K.-F.-Friesenstr. 26 ★
★4690 Herne 1 · 02323/40887★

ORIG. GFA-DRAFT 250 DM
nach 18 Uhr. Tel. 0202/400182

★★★ CW-CHART ★★★
Börsenpaket Charts+Depot ab 99
Demo 40,– Info C. Weisenböhler
Waldstr. 2, 7175 Vellberg
★★★ CW-DEPOT ★★★

GfA-Basic-Compiler(!), Original,
DM 120,–. Tel. 040/6472427

GFA-Basic V2 neu Original 125
DM GFA-Compiler 125 DM. Weitere Original Software.
H. Schaber, Industriestr. 4a,
7707 Engen, 07733-8401

Atari ST Public Domain Service
Für nur 17 DM 7 Disks im Dez u.
Jan (incl. Porto) also 2,40 DM pro
Disk. Info (Rückumschlag) bei:
R. Rieckmann, Hauptstr. 33, 3101
Langlingen (kein Abo!)

Megamax ST-Pascal+ 05641/8754

Einkommensteuer/Lohnsteuer-Jahresausgleich **jetzt 1986 u. 87**
GFA/20 DM Vorkasse incl. Disk
u. Porto/Jochen Höfer, Grünewald
2a, 5272 Wipperfürth/02267/3368

Hübsche Mädchen, Raumschiffe
und vieles mehr biete ich auf über
80 Disketten!!! Gratisinfo bei Ralf
Markert, Balbachtalstr. 71, 6970
Lauda-Oberbalbach

Public-Domain-Sammlung pro Disk
(doppels.) 8 DM, Liste anfordern.
Floppy 354SF 200 DM, GFA-Compiler 140 DM, Protext ST
100 DM, K.-D. Prochaska
0231/215813

Megamax C Compiler orig. 399,-
V 1.1 neu. Tel. 08669/12337

★★★ **PD-SOFTWARE** ★★★
Vielseitiges Angebot für ATARI ST
(Anwendungen, Sprachen, etc) für
jeden etwas, Liste gegen frank.
Rückumschlag, H. Rode, Langreder
Str. 1, 3 Hannover 91

★ **Super Software aus USA** ★
LDW Basic Compiler 159,– DM
Neu: Adventures in Deutsch
Phantasie1 79,– in Kürze:
Phantasie2, Rings of Zilfin um.
Bitte Katalog anfordern: **DDV**
208 Pinneberg Thesdorferw. 169a

---

100 Disketten PD-Software für den
ST sind in unserem neuesten, komplett überarbeiteten und erweiterten
Katalog enthalten! Alle Programme
auch auf zweiseitigen Disketten extrem preiswert erhältlich, mit Programmbeschreibung und vielen
Sonderangeboten. Außerdem verschiedenste Soft- und Hardwareangebote zu Tiefstpreisen. Wo? Bei
Gerald Köhler, Mühlgasse 6, 5991
Igersheim, Tel. 07941/44661.

PUBLIC DOMAIN PROGR.
auf Markendisk (10) nur 79,- DM
zuzügl. Porto
Inge Dander, Rockefellerstr. 66
8 München 45 · Tel. 089-3111782

Überflüssige Weihnachtsgeschenke
günstig abzugeben:
The Pawn DM 80,– · Rogue, Borrowed Time, Phantasie, Cards je
70,– · PowerPlan Spreadsheet mit
Grafik DM 95,– · AssemPro 68000
dev. Package DM 85,–
K. Schultheis, Tel. 06151-594212

Public Domain auf Markendisk.
★Paket 10 Stck. **nur 75,– DM**★
★auch die PD v. ST Computer★
★jede Menge neue Programme★
★ neue Serviceleistungen ★
★ !! WIR BIETEN MEHR !! ★
★ **Fordern Sie die neue Liste** ★
★ Software Inge Dander ★
★ Rockefellerstraße 66 ★
★
8000 München 45 ★
★ Tel. 089/3111782 ★

Wir sind zwar nicht die einzigen
Anbieter von PD-Software für den
ST, aber vielleicht sollten Sie doch
einmal unseren Katalog anfordern.
Schreiben Sie an Gerald Köhler,
Mühlgasse 6, 6991 Igersheim,
Tel. 07931/44661 (24 h).

The Pawn
– Die Lösung –
auf IDL P/D-Disk

● **Public Domain Software** ●
Riesiges Angebot, Kopiere **nur einzelne Programme**, natürlich ein- und doppelseitig in allen Formaten!
Nur 4,50 DM je 360 KByte! Info
bei Eckart Stinshoff / Hökholz /
2336 Waabs

Fractal-Generator, sehr schnell ca.
48 000 Multiplikationen/Sek. d.
Maschspr. inkl. Animation RGB u.
MONO. Rolf Kordts, Mathildenstr.
17, 2217 Kellinghusen 69,–

Videofilmverwaltung und komf.
Prg. z. Textfileausdruck in versch.
Schriften: je nur 29 DM, zus.
49 DM, Best. + Info: 0661/67362

Berlin umfangreiches PD-Software-Angebot pro Disk 8 DM (incl. Disk) Info anfordern 7118941

Platine-ST orig. VB 350,– (NP: 698,–) 07543/3974 ab 18 h

PD-SOFTWARE
Kopierkosten 2,50 DM/Disk!
Leerdiskette 4,00 DM !
Gratisinfo bei E. Twardoch, Kaiser-Wilhelm-Str. 88, 1 Berlin 46

Publik Domain + Disk 9 DM incl. Nk. **Info** m. Freiumschlag anf. 10 **3,5" Disk** 39 DM. **Atariuhr** i. Modulp. 109,– **Fa. Th. Bandt, Altenbergring 7, 6991 Igersheim 07931/42117**

---> **Vereinsverwaltung** <---
Freie Gestaltung von Mahnungen und Beitragsabrechnungen.
Disk u. Dokumentation 80,-- DM
V. Reichel · Durlacherstraße 23
7500 Karlsruhe 41

P6/P7-User: Der FX80-Emulator, ein Treiber für Grafik-Ausg., incl. Hardcopy b&w und color!
Info: 0911/767851 ab 18 Uhr.

Für Lehrer, Prüfer usw.:
Testauswertung u. Namensdatei
Bewertung aller Aufgaben und Schüler mit Ausdruck. Diskette gegen 40 DM bei Helmut Geiger, Wiesenstr. 13, 7250 Leonberg 7

VIP-Tab. Kalk. (Org.) deutsch DM 590,–, Side Click 80,– DM + Sonst. Tel. 08084/2633 ab 18⁰⁰

GFA-Basic V20. Tel: 05139-3636

PD-Software für den ST? Wir sind die Spezialisten mit dem größten Programmangebot (> 100 randvolle Disketten)! Viele weitere starke Angebote für Hard + Software rund um den Atari ST (Diskbox für 80 Disketten 24 DM Quickshot II 18 DM)! Gratiskatalog anfordern bei Gerald Köhler, Mühlgasse 6, 6991 Igersheim, Tel. 07941/44661.

Computerviren? (s.c't3/87)
ATOX schützt **Atari ST** Disk. Quellcode g. 20 DM Unkostenbeteiligung a. D. Krabel, Urbanstr. 47, 1 Berlin 37

Lotto-Programm
F. Klein, Dessauer Str. 5a, 8900 Augsburg 21, Tel. 0821/812157
Preissenkung: nur noch **39 DM.** Auf Wunsch 20 seit. Dokumentation vorab geg. 2,40 in Briefm.

Von Lehrer für Lehrer! Praxiserpr. Notenverwaltungsprogramm voll GEM unterstützt mit über 10 versch. Funkt. (Gratisinfo geg. fr. Rückumschlag) für DM 40,–
H. J. Merkel, Nahestr. 28, 66 SBR

Berlin: Public Domain für Atari ST. 50 Disks je DM 8,–, alle für 350,– inkl. Disk. Anrufen, hingehen, mitnehmen. Tel: 8254163

PD-Software? Kein Problem!
Massenweise + billig für Ihren ST!
Gratisinfo bei R. Markert, Balbachtalstr. 71, 6970 Lauda

FE-ST 3D-Finite Elemente
für statische Berechnungen
Dipl. Ing. H.-P. Wendorff
Burgstr. 5
3008 Garbsen 5

CH – Atari ST Software – **CH**
Verkaufe Original-Software inkl. Handbücher: **K-Spread** SFR 75. **K-Graph** SFR 75 **Boffin Text** SFR 150. **The Pawn** SFR 40. Info bei: M.Schmid, Tel. 041/454609 ab19⁰⁰

Je 1x Orig. Progr. + Handbuch zum halben Preis:
Becker Datamat 50 DM
Becker Textomat 50 DM
MICA-CAD 150 DM
Beer, Tel. 02371-61787

1st-Proportional Proportional Druck im Blocksatz m. 1st Word f. NEC P6/7, HR15u.a. Prg (GEM) m. deut. Handb. 69 DM (incl. Porto/V.)
Kniß, 5100 Aachen, Viktoriastr. 9

BAS-Scheck für alle ATARI-ST's GEM-unterstütztes Programm z. Verwaltung und zum Druck von Überweisungen! Info: 02245/2740

Großes Angebot an PD-SOFT für ST vorhanden. Fordern Sie Gratisinfo an bei K. Galz, Sonnenbergstr. 55, 7800 Freiburg

---

Verkaufe: **SM-Text** 140,-, **Adress Perfekt** 80,-, **Datamat ST** 60,-, **DB-Calc** 80,- ab 18⁰⁰ (04129-452)

Textverarbeitung: SM-Text 100 DM
Adventure: The Pawn 50 DM; Nolte, 7817 Ihringen, Wasenweilerstr. 11a

---

Verk. 1st. Lektor + Side Click: 200,– Tel. 02842/1782

ST: **Super-Plotteremulator** für FX80 u. Kompatible, Disk DM 45
Ultrabillig... Für nur DM 49,– kopieren wir Ihnen die besten aller PD-Prg. (über 200) auf 21 Disketten. Bitte senden sie uns nur SS-form. Disks. Zahlungen per V-Scheck. M. O. Stehr, Ahornweg 7 2409 Scharbeutz 1

KONTAKTE

Suche ST Kontakte im Hochheimer Raum und suche Tauschpartner für ST-Programme. M. Theimer, Schwedenstr. 2, 6203 Hochheim

---

ST-User sucht Kontakte
zwecks Erfahrungstausch!
Tel.: 0209/379572

Gesucht Programmierer auf Atari ST. Bereich 3D Grafik/Sound/Monochrom/Stimmen/Erotik! Können Sie Drehbuch-Progr. umsetzen? Progr. müssen 100 % fehlerfrei laufen. Raum Mittel-Süd-BRD und CH! Bitte Offerte mit evtl. Test-Prog. und Std.-Honorar an SWISS VIDEO/Software Eros · Postfach · CH-8040 Zürich

---

C. Pascal, Forth, Games, Business, u.v.a. ab **DM 2/Disk.**
Info gegen Rückporto!
Michael Horak, Baumeisterweg 12, 7000 Stuttgart 1

TAMAS-THEATER sucht ST
Programmierer/Elektroniker
zw. f. Mitarbeit. Tel. 07777-1461

Modula II Interessierte aus ganz Deutschland gesucht.

Atari Freaks zum Interessenaustausch gesucht.
Braun Ludwig jun., Schwaigerstr. 4, 8425 Neustadt-Geibenstetten

TAUSCH

■Tausche LATTIC C Compiler
■gegen Easy Draw oder Mica CAD
■Kurt Oertli, Untere Allmeind 4
■CH-8755 Ennenda, Schweiz

VERSCHIEDENES

NEU! NEU! Atari APDCG Club!
Kontaktservice, Diskmagazin, PD, Infodisk gegen 7 DM!
Wer hat Lust bei uns mitzumachen?
Frank, Carl-Spitzweg 17,
7920 Heidenheim 5

Wer hat Erfahrung mit **Emuf-Heizung** 040-5514960, 04193-3701

Suche ORG. PRG's P.D. usw. Tauschpartner u. neue Software!
Tel.: 0209/379572

Verkaufe Protext, 120 DM.
Suche komplette SF 354.
Tel. 08633/1407

März '87 – Anzeigenschluß am 3. Februar 1987

April '87 – Anzeigenschluß am 3. März 1987

Verspätete Einsendungen kommen in die darauffolgende Ausgabe!

68 000er-Bücher für den Atari ST



An den Assembler-Programmierer richtet sich eine Reihe von Büchern, die sich mit dem 68000er-Prozessor von Motorola beschäftigen. Freilich sind nicht alle Produkte aus diesem reichhaltigen Angebot brauchbar. Den Anfang unserer Berichterstattung machen vier Bücher über den 68000er. Zwar können wir im Rahmen dieser Buch-Ecke nicht über alle Bücher zu diesem Thema berichten, doch werden wir auch in zukünftigen Ausgaben Veröffentlichungen über den 68000er behandeln.

Die Standardwerke

Als regelrechtes Standardwerk kann man die zwei Bände mit dem Titel **M68000 Familie** (Teil 1 und Teil 2) des TE-WI-Verlages in München bezeichnen: Ein rundum gut durchdachtes Produkt der Autoren Werner Hilf und Anton Nausch. Es liefert alle nur erdenklichen Informationen, die Entwickler im Hard- und Softwarebereich benötigen. Der Leser erfährt so manchen Profi-Trick und kann sich mit dieser Hilfe einiges zusammenreimen. Interessantes für den Anwender des Atari ST gibt es eine ganze Menge – obwohl das Buch nicht speziell auf diesen Rechner zugeschnitten ist. Für Anfänger sind die beiden Bände freilich nur mit Einschränkungen zu empfehlen, da einiges Wissen vorausgesetzt wird. Weitere Informationen können Sie in der Buchbesprechung im Heft 11/86 dieser Zeitschrift nachlesen.

Werner Hilf und Anton Mausch:
M68000 Familie
Teil 1 und Teil 2
TE-WI-Verlag
8000 München 40
ISBN 3-921803-16-0

Auch ein Standard

Ein ganz ähnliches Konzept liegt dem von Data-Becker verlegten Prozessor-Buch zum 68000er zugrunde, in dem der Autor Grohmann Eichler den inneren Aufbau und die Programmierungstechnik sehr genau beschreibt.

Zu Beginn informiert das Buch über die Entwicklung des leistungsstarken Prozessors und über seine innere Architektur. In diesem Hardware-Teil werden die verschiedenen Signaltypen und die Bus-Organisation beschrieben, daneben gilt das Augenmerk anderen Mitgliedern dieser Prozessor-Familie und den bekanntesten Peripheriebausteinen. Zweiter großer Schwerpunkt des Buches ist die Programmierung des Prozessors. In diesem Teil sind alle Adressierungsarten aufgeführt und sämtliche Befehle (auch für den 68020) beschrieben.

Einziger Minuspunkt: Die konkreten Beispiele sind ein wenig mager ausgefallen. Für Anfänger ist auch hier Vorsicht geboten.

Grohmann Eichler:
Das Prozessor Buch zum 68000
Technik und Programmierung

Data-Becker-Verlag
Merowingerstraße 30
4000 Düsseldorf
ISBN 3-89011-094-0

ATARI speziell

Die speziell dem Atari ST gewidmeten 68000er-Bücher sind nicht gerade zahlreich. Da das Betriebssystem auch von höheren Sprachen (etwa C, PASCAL, MODULA oder BASIC) angesprochen werden kann, ist die Anwendung von Assembler nicht immer sinnvoll. Für eingefleischte Programmierer und solche, die es werden wollen, bleibt es jedoch immer ein Reiz, das Innere eines Rechnersystems zu erforschen.

Schon seit längerer Zeit gibt es beim Data-Becker-Verlag ein Buch, das sich hauptsächlich mit der Atari-Assembler-Sprache beschäftigt. Auch hier steht eine kurze Erklärung des 68000er Prozessors am Anfang; die späteren Kapitel konzentrieren sich auf das Betriebssystem des Atari ST.

Die Benutzung der Routinen GEMDOS, BIOS und XBIOS wird an vielen Beispielen erläutert. Der Anfänger wird vom Schreiben allgemeiner Programme bis hin zum Einbinden systeminterner Routinen so gut unterrichtet, daß er nach kurzer Zeit selbst in der Lage ist, Programme in Assembler zu schreiben. Zudem befaßt sich das Buch eingehend mit der Bedienung des Assemblers von Digital Research und erklärt ausführlich alle Schritte von der Editierung eines Programms bis zur Erstellung einer lauffähigen Version.

Grohmann/Seider/Slibar:
Das Maschinensprachebuch
zum Atari ST
Data-Becker-Verlag
ISBN 3-89011-120-3

Assembler-Praxis

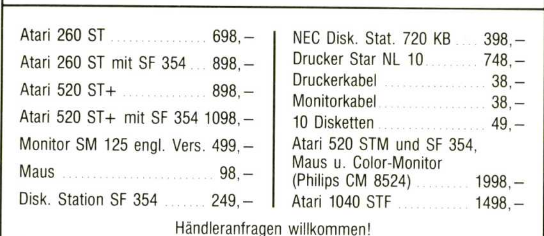
Mit dem Titel „Assembler Praxis auf ATARI ST“ bietet der TE-WI-Verlag ein Buch, das sich fast ausschließlich mit der Programmierung des ST-Betriebssystems beschäftigt. Nach der einführenden Beschreibung der Merkmale des 68000-Prozessors liefert der Autor Roland Löhr eine kurze Beschreibung verschiedener Assemblersprachen. Eine ausführliche Erklärung der Betriebssystems-Routine vermittelt dem Anwender das notwendige Basiswissen, mit dem er die Beispiele vollständig nachvollziehen kann.

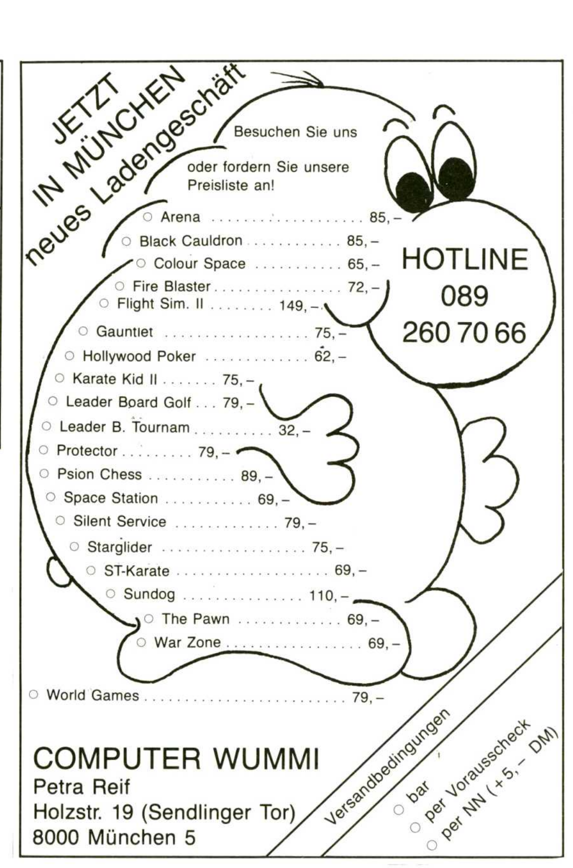
Verschiedene Programme zur Demonstration der Wirkung von GEMDOS, BIOS und XBIOS beschließen das Buch.

Roland Löhr:
Assembler Praxis auf ATARI ST
TE-WI-Verlag
ISBN 3-921803-70-5

COMPUTERVERSAND
WITTICH
Tulpenstr. 16 · 8423 Abensberg
☎ **0 94 43 / 453**







○ World Games 79,–

COMPUTER WUMMI
Petra Reif
Holzstr. 19 (Sendlinger Tor)
8000 München 5

Versandbedingungen
○ bar
○ per Vorausscheck
○ per NN (+5,– DM)



CONKIT der RSC-Editor
ob Sie Ihre eigenen RSC-Files erstellen oder nur Ihre englischen Programme mit deutschen Texten (natürlich geht das!!!) versehen wollen. CONKIT ist die ideale Hilfe dafür. Das Programm beinhaltet einen Icon- und Window-Editor, einen Window-Manager als Assembler-Source (sehr komfortabel) sowie ein Accessory, durch das Bildschirmausschnitte in das RSC-File übernommen werden können.
Das alles für nur **DM 74,–**



COMPRESS der Datenkompressor
Schreiben Sie viel oder haben Sie große Dateien oder benutzen Sie einen Compiler? Dann kennen Sie das Problem mit dem Diskettenwechseln. Wäre es nicht schön, wenn alle Texte zusammen mit der Textverarbeitung auf eine Disk passen würden? Dies ist nun möglich. Vergleichen Sie selbst.
Durchschnittliche Einsparung bei normalen Texten: 45%, bei Assemblersource: 55% und selbst bei Programmen werden noch 25% gespart. Und das bei einer sensationellen Verarbeitungsgeschwindigkeit von 15 000 Byte Text in weniger als 1 Sek.
Der Preis: **DM 66,–**

Bestellungen per Vorauskasse oder Nachnahme (+5 DM) oder weiteres Info (Rückporto) bei:

Peter Kull Software-Entwicklung
Kirchtalstr. 29 · 7000 Stuttgart 40 · Tel. 07 11 / 87 26 08

DR. ZOCK ähnlich **Break out**
Superschnelle Arkade nur 19,90 DM

JUMPER ähnlich Qubert
Spielespaß ohne Ende nur 19,90 DM

Beide Spiele nur für Colour-Monitor

Liste **nur** gegen Freiumschlag

CSM
Computer-Spezial-Marketing GmbH
Holzstr. 19 · 8000 München 5
Tel.: 0 89 - 260 70 66

Lieferung: Scheck oder Vorauskasse
Bei Nachnahme zuzügl. 3,– DM Versandskosten

★ ★

★ **PRG für ATARI ST – EXZELLENT IN STRUKTUR, GRAFIK, SOUND** ★
★ **– ALLE PRG IN DEUTSCH – ALLE PRG IN S/W UND FARBE** ★

★ **GELD** – 25 Rechenroutinen mit Ausdruck für Anlage – Sparen – Vermögensbildung – Amortisation – Zinsen (Effektiv-/Nominal-) – Diskontierung – Konvertierung – Kredit – Zahlungsplan usw. 96,– ★

★ **GESCHÄFT** – Bestellung, Auftragsbestätigung, Rechnung, Lieferschein, Mahnung 6 Briefrahmen mit Firmendaten zur ständigen Verfügung (Anschrift, Konten usw., Menge/Preis, Rabatt/Aufschlag, MWst., Skonto, Verpackung, Versandweg usw.) 96,– ★

★ **ETIKETTENDRUCK** ' bedruckt 40 gängige Computer-Haftetiketten-Formate nach Wahl und Auflagebestimmung, kinderleichte Gestaltung, Ablage für wiederholten Gebrauch 89,– ★

★ **ASTROLOGISCHES KOSMOGRAMM** – Nach Eingabe von Namen, Geb. Ort (geogr. Lage) und Zeit werden errechnet: Siderische Zeit, Aszendent, Medium Coeli, Planetenstände im Zodiak, Häuser nach Dr. Koch/Schäck (Horoskop-Daten m. Ephemeriden) – Auch Ausdruck auf 2' DIN A4 mit allgemeinem Persönlichkeitsbild und Partnerschaft 75,– ★

★ **BACKGAMMON** – überragende Grafik, gänzlich mausgesteuert, ausführliche Spielanleitung, lehrreiche Strategie des Computers, in 6 Farben bzw. Grauabstufungen bei S/W 58,– ★

★ **CASINO-Roulett** – Mit Schnellsimulation, Chancentest, Sequenzenverfolgung, Kassenführung 68,– ★

★ **KALORIEN-POLIZEI** – Nach Eingabe von Größe, Gewicht, Geschlecht, Arbeitsleistung erfolgt Bedarfsrechnung und Vergleich m. d. tatsächlichen Ernährung (Fett, Eiweiß, Kohlehydrate), Idealgewicht, Vitalstoffe, auf Wunsch Ausdruck 56,– ★

★ **BIORHYTHMUS** – zur Trendbestimmung des seelisch-/geistig-/körperlichen Gleichgewichts, Zeitraum bestimmbar – Ausdruck mit ausführlicher Beschreibung über beliebigen Zeitraum 56,– ★

★ **FONT EDITOR unter DEGAS:** 12 bekannte Schriftarten m. deutschem Zeichensatz 64,– ★

★ **Bibliothek** – schreibt Listen oder Etiketten, auch auszugsweise nach codierten Kriterien (Titel/Best.-Nr./Verfasser/Verlag/Gruppe/Bestand/Preis/usw.) 86,– ★

★ **GEM + MERGE:** Die GEM (AES + VDI)-Applikationen unter ST Basic – ca. 100 Routinen zum Mergen für Funktion und Aufruf – Führt zur GEM-Beherrschung 58,– ★

★ **usw. usw. – Fordern Sie mit Freiumschlag unsere Liste an!** Im Computer-Center oder bei uns zu obigen, unverbindlich empfohlenen Preisen + DM 3,– bei Vorkasse oder DM 4,70 bei Nachnahme. ★

★ **I. DINKLER, IDEE-SOFT,** Am Schneiderhaus 7 ★
★ D-5760 Arnsberg 1 · Tel.: 0 29 32 / 3 29 47 ★

★ ★

+ + stop + + + neues zum thema ST + + + stop + + +

1 **MTV-1** – Die SM-124/Farbmonitor-Umschalteinheit
Endlich Schluß mit der ewigen, lästigen Stöpselei!

2 **JOY-1** – Das Interface für Joysticks als Mausersatz
Preiswerte und platzsparende Alternative zur Maus!

3 **BUS-1** – Die Low-Cost-Anschlußerweiterung
Erweitern Sie einfach die E/A-Möglichkeiten Ihres ST!

...und der Preis? Sensationell: jedes Interface nur 48 DM!

Versand per Vorkasse (+ 2,50 DM) oder Nachnahme (+ 7,00 DM)

Weiteres Zubehör (Software, Drucker, Disketten, etc.) bitte anfragen (2 DM in Briefmarken beilegen).

– Händleranfragen erwünscht!

Wir machen Spitzentechnologie anwender-freundlich.



Postfach 579
D-4600 Dortmund 1
☎ 02 31 / 57 74 62

Das lesen Sie unter anderem in der Märzausgabe der ST Computer:

CeBit '87 – Was wird man neues sehen

Alle Neuigkeiten rund um ATARI, die zu sehen sein werden

512 KByte mehr ROM-Speicher

Wir stellen unsere neue ROM/EPROM-Karte zum Easyprommer vor

Fortran 77

Eine der ältesten Computersprachen, die jahrelang ein muß für Lösungen im naturwissenschaftlichen Bereich war, gibt es auch für den ATARI ST.

Der ST kommuniziert mit der Außenwelt

Wir geben Ihnen das Knowhow, um unseren 16 Bit Eingabe/Ausgabe-Port, der am Harddisk-Anschluß betrieben wird, nachzubauen.

Alle Programmiersprachen für den ST

Eine Zusammenfassung aller zur Zeit erhältlichen Programmiersprachen mit Preis und Bezugsquelle.

Forth – die Sprache der vierten Dimension

Wir starten unseren neuen Kurs in der Programmiersprache Forth. Die benötigten Werkzeuge (Interpreter/Compiler) sind kostenlos (Public Domain Nr. 3 und 4).

Künstliche Intelligenz

Um den Einstieg in die KI zu vereinfachen, beginnen wir mit dieser neuen Serie. Sie wird anhand ausgewählter Beispiele die Arbeitsweise der KI-Sprachen verdeutlichen. Als Grundlage finden Sie zwei Sprachen (XLISP und PROLOG 10) in unserem Public-Domain Service.

Die Märzausgabe erscheint am 27. Februar 1987

Inserentenverzeichnis

ST-Computer – Einzelheft-Bestellung

ST-COMPUTER können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 6,– (zuzüglich Gebühr für Porto u. Verp.) nachbestellen.
Bearbeitung **nur gegen beigefügten Scheck** über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli/Aug.	Sept.	Okt.	Nov.	Dez.	
											1986 = DM
											1987 = DM

+ Gebühr für Porto u. Verp. = DM

☐ Scheck in Höhe zus. DM ________ liegt bei

Vorname/Name ________________________________

Straße/Hausnr. ________________________________

Ort ________________________________

Datum/Unterschrift ________________________________

Gebühr für Porto u. Verpackung: 1 Heft DM 2,–; 2 bis 6 Hefte DM 3,–; ab 7 Hefte DM 5,–

Heim-Verlag · Heidelberger Landstraße 194 · 6100 Darmstadt-Eberstadt

Impressum

ST-Computer

Herausgeber: Heim Fachverlag, Heidelberger Landstraße 194, 6100 Darmstadt 13, Telefon (0 61 51) 5 60 57

Verlagsleitung: Hans-Jörg Heim

Redaktion: Uwe Bärtels (UB-Chefredakteur). Harald Egel (HE) Marcelo Merino (MM), Harald Schneider (HS) Schwalbacher Str. 64, 6236 Eschborn, Tel. 0 61 96/48 21 58,

ST-Computer Redaktion, Postfach 59 69, 6236 Eschborn

Redaktionelle Mitarbeiter: Markus Nerding (MN), Oliver Joppich (OJ), Uli Eickmann (UE) Jürgen Leonhard (JL)

Autoren dieser Ausgabe: H. Bauch, P. Gebhart, S. Höhn, O. Joppich, Dr. Sarnow, C. Schormann, S. Schuler, G. Viehöver, T. Weinstein, J. Wilhelm, K. Wolf

Titelseite: M. Fabian

Produktion: Klaus Schultheis (Ltg.), Bernd Failer, Susanne Failer, Bela Kumar, Carola Schwarze, Martina Simacek

Anzeigenverkaufsleitung: Uwe Heim

Anzeigenpreise: nach Preisliste Nr. 2 gültig ab 1.7.86

Vertrieb: Hans-Jörg Heim, Uwe Heim, Heide Schultheis

Erscheinungsweise: 11 x jährlich

Bezugspreis: Einzelheft DM 6,–. Jahresabonnement DM 60,– inklusive der gesetzlichen Mehrwertsteuer und den Zustellgebühren für 11 Ausgaben. Ausland: 80,– DM inkl. Versand

Bezugsmöglichkeiten: ATARI-Fachhändler, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag unter obiger Adresse.

Druck: Ferling Druck Darmstadt

Manuskripteinsendungen: Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit ihrer Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern im Heim Verlag. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

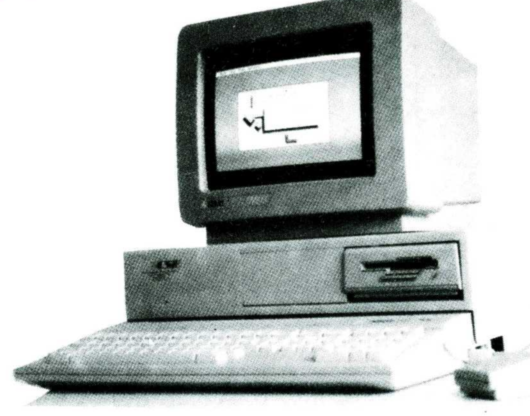
hardware
software
organisation
service

CSF GMBH

Heeper Str. 106–108, 4800 Bielefeld 1, 0521/61663

Kein Kabelsalat mehr mit dem Gehäuse für ATARI ST





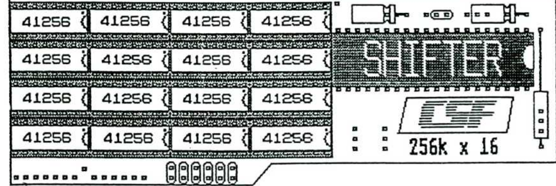
- **Zentrale Stromversorgung für alle Geräte einschl. 2 Drucker**
- **Einbaumöglichkeit von 2 Diskettenlaufwerken**
- **Rechner (Tastatur) kann komplett unter das Gehäuse geschoben werden (Staubschutz)**
- **Massives Blechgehäuse**

ATARI ST-Gehäuse erhalten Sie bei den autorisierten Fachhändlern

NEUES VON CSF – Speicherweiterung ohne Probleme!

— **steckbar** (ohne jegliche **Lötarbeiten**) — läuft auch auf dem 520 STM
— enorme Zeitersparnis durch einfache, bebilderte Einbauanleitung
— kein Flimmern nach der Erweiterung (durch separate, geglättete Spannung an der zweiten RAM-Bank)
— sensationeller Preis
— Bei Bestellungen bitte angeben:
☐ Speichererweiterung steckbar
☐ Speichererweiterung zum Löten

DM 225,—



GEHÄUSE FÜR ATARI 1040 STF

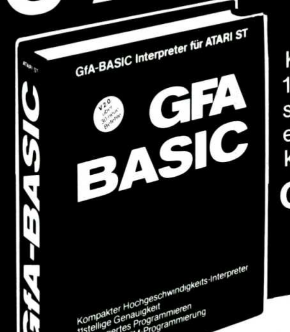
DM 198,—

In Kürze lieferbar. (Bestellungen bereits möglich)

Alle Preise sind unverb. empf. Verkaufspreise

Zu beziehen:	In der Schweiz:	In Österreich:
Direkt bei CSF, Bielefeld **Tel. 0521/61663** **Bei allen ATARI-Händlern**	**SENN** **Computer AG** **Langstr. 31 · CH-8021 Zürich** **Tel. 01-241 73 73**	**Institut für Datenverarbeitung** **und Organ. Ges.mbH** **Rehberger Hauptstr. 95 · A-3503 Krems** **Tel. 02732-70581**

Für alle ATARI ST



Kompakter Hochgeschwindigkeits - Interpreter, 11stellige Genauigkeit, strukturiertes Programmieren, einfachste GEM-Programmierung, komfortabler Editor.

GFA-BASIC Interpreter V 2.0
DM 169,–

GFA-BASIC Compiler für ATARI ST
GFA BASIC
GFA-BASIC

GFA-BASIC Compiler
DM 169,–

Schneller 2-Pass-Compiler, benötigt keinen Linker, erzeugt sehr schnelle, kurze Programme ohne Runtime-Modul, voll kompatibel zum Interpreter, einfachste Bedienung.

GFA-Buch ATARI ST
GFA BASIC
GFA-BASIC

GFA-Buch DM 79,–
Incl. Diskette mit Beispiel-Programmen
Lieferbar voraussichtlich Ende Februar 87



Schnelles, interaktives 3D-Grafik-Programm zum Generieren von 2D- und 3D-Objekten.

Aus GFA-BASIC heraus können problemlos 2D- oder 3D-Spiele, Animationen oder bewegte Simulationen erzeugt werden.

GFA-VEKTOR DM 149,–



Voll GEM-gesteuertes, leistungsfähiges CAD-Programm, maßstabsgerechtes Erstellen von Zeichnungen in Zoll und mm, Bildausschnitte und Symbole beliebig manipulierbar und gradweise drehbar, mächtige Zeichenfunktionen wie z. B. Lot, Winkel zu Geraden, Kreis durch 3 Punkte, etc. 255 Zeichenebenen (10 gleichzeitig darstellbar). Schraffieren von beliebigen Flächen. Einfachste Handhabung. Läuft auf fast allen Plottern und Druckern. Symbole und Bibliotheken in beliebiger Menge anlegbar.

GFA-DRAFT DM 298,–

...Anruf genügt: 02 11-58 80 11

GFA Systemtechnik GmbH

Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 02 11/58 80 11

